Stochastic Realization Theory for Exact and Approximate Multiscale Models

by

Dewey S. Tucker

B.S., Electrical and Computer Engineering
Georgia Institute of Technology, 1995

S.M., Electrical Engineering and Computer Science
Massachusetts Institute of Technology, 1997

Submitted to the Department of Electrical Engineering and Computer Science in partial fulfillment of the requirements for the degree of

Doctor of Philosophy
in Electrical Engineering and Computer Science
at the Massachusetts Institute of Technology

September 2005

Signature of Author: ______________________________
Department of Electrical Engineering and Computer Science
August 30, 2005

Certified by: ______________________________
Alan S. Willsky
Edwin Sibley Webster Professor of Electrical Engineering
Thesis Supervisor

Accepted by: ______________________________
Arthur C. Smith
Professor of Electrical Engineering
Chair, Committee for Graduate Students

Stochastic Realization Theory for Exact and Approximate Multiscale Models

by Dewey S. Tucker

Submitted to the Department of Electrical Engineering
and Computer Science on August 30, 2005
in Partial Fulfillment of the Requirements for the Degree
of Doctor of Philosophy in Electrical Engineering and Computer Science

Abstract

The thesis provides a detailed analysis of the independence structure possessed by multiscale models and demonstrates that such an analysis provides important insight into the multiscale stochastic realization problem. Multiscale models constitute a broad class of probabilistic models which includes the well-known subclass of multiscale autoregressive (MAR) models. MAR models have proven useful in a variety of different application areas, due to the fact that they provide a rich set of tools for various signal processing tasks. In order to use these tools, however, a MAR or multiscale model must first be constructed to provide an accurate probabilistic description of the particular application at hand. This thesis addresses this issue of multiscale model identification or realization.

Previous work in the area of MAR model identification has focused on developing algorithms which decorrelate certain subsets of random vectors in an effort to design an accurate model. In this thesis, we develop a set-theoretic and graph-theoretic framework for better understanding these types of realization algorithms and for the purpose of designing new such algorithms. The benefit of the framework developed here is that it separates the realization problem into two understandable parts – a dichotomy which helps to clarify the relationship between the exact realization problem, where a multiscale model is designed to exactly satisfy a probabilistic constraint, and the approximate realization problem, where the constraint is only approximately satisfied.

The first part of our study focuses on developing a better understanding of the independence structure exhibited by multiscale models. As a result of this study, we are able to suggest a number of different sequential procedures for realizing exact multiscale models. The second part of our study focuses on approximate realization, where we define a relaxed version of the exact multiscale realization problem. We show that many of the ideas developed for the exact realization problem may be used to better understand the approximate realization problem and to develop algorithms for solving it. In particular, we propose an iterative procedure for solving the approximate realization problem, and we show that the parameterized version of this procedure is equivalent to the well-known EM algorithm. Finally, a specific algorithm is developed for realizing a multiscale model which matches the statistics of a Gaussian random process.

Thesis Supervisor: Alan S. Willsky
Professor of Electrical Engineering and Computer Science

Acknowledgments

> ... And we rejoice in the hope of the glory of God. Not only so but we also rejoice in our sufferings, because we know that suffering produces perseverance; perseverance, character; and character, hope. And hope does not disappoint us, because God has poured out his love into our hearts by the Holy Spirit, whom he has given us.
> — *Romans 5:1-5*

> In order to compose, all you need to do is remember a tune that nobody else has thought of.
> — *Robert Schumann*

Never could I have imagined that this journey would present so many challenges. I am standing here at the end of this journey because I have had wonderful guidance and support from the people I have met along the way. It is to these people that I would like to now say "Thank you".

First, I would like to thank my guide and mentor along this journey, my advisor Professor Alan Willsky. I have known about Alan ever since I took my first Signals and Systems course using his book, and I knew then that I would like to work for him. Little did I know that I would have the opportunity to assist him in teaching the Signals and Systems course at MIT. I thank Alan for providing monetary support, for his constant feedback and constructive criticism, and for the skills I have gained just by working with him.

I would also like to thank my committee members, Professors Munther Dahleh, Hanoch Lev-Ari, and George Verghese. Each of these men is a distinguished and respected member of their field of research, and I am honored to have had them serve on my committee. They have been supportive in a number of different ways – providing letters of reference, through classroom interactions, and by providing feedback on the ideas covered in this thesis. I would be remiss if I did not also thank the National Science Foundation for their fellowship which provided three years of funding and allowed me to freely focus on my studies.

I have also had the privilege to work with a number of bright research scientists and post-doctoral students who have been part of the Stochastic Systems Group (SSG) at one time or other. Among these are Hamid Krim, John Fisher, Khalid Daoudi, Müjdat Çetin, Anthony Yezzi, and Jun Zhang. Each of them provided his assistance and expertise to the group and to me.

During my stay in SSG, I have known a number of graduate students who in one way or another have contributed to my success. My predecessors in this group were instrumental in providing guidance when I first began my Ph.D. I would especially like to thank Bill Irving, Mike Daniel, Seema Jaggi, Paul Fieguth, Cedric Logan, Terrence Ho, Ilya Pollak, Austin Frakt, and Mike Schneider. I have also had a number of contemporaries who entered the group or MIT around the same time that I did. I enjoyed the times we spent together studying, teaching, and talking about our research ideas. Among these people, I would like to thank Nick Laneman, Pat Kreidl, Martin

Wainwright, Jason Johnson, Taylore Kelly, John Richards, Shane Haas, and Ron Dror. Finally, I would like to thank those students who joined SSG after I joined: Junmo Kim, Lei Chen, Jason Williams, Erik Sudderth, Ayres Fan, Dmitry Malioutov, and Alex Ihler. I wish all of you success in your future endeavors, and I thank you for your friendship.

In the preceding list of names, I have intentionally left out three people who I have known for a number of years, and who I consider to be among my best friends. They are Andrew Kim, Walter Sun, and Andy Tsai. When I was down and dejected, these three always knew how to cheer me up and how to encourage me to finish this journey. I owe them a very special thank you.

While the aforementioned people have been instrumental in my journey at MIT, there are a number of people who have watched over my spiritual journey. I would like to thank all of the members of First Baptist Church in Fitzgerald and Tremont Temple Baptist Church in Boston for their constant prayers of support. I would specifically like to thank Blane and Barbara Jacobs, Gene and Pat Wilder, Susan Spivey, Marty Smith, Dick and Vonnie Mann, Dan and Cathy Foster, Joan McDonald, Ray Pendleton, John Kennedy, and Donna Currie. And, my greatest Guide in this journey, and in all of the journeys I will undertake, is my Lord and Savior Jesus Christ with whom everything is possible.

Finally, I would like to thank all of my friends and family who have taken such a personal interest in my success. I would especially like to thank my parents, JoAnn and Dwight, who have been so understanding throughout this entire process. I know it has not been easy, but I thank you for your unwavering support and constant love.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

THE goal of this thesis is to illuminate and investigate a number of important ideas related to the multiresolution probabilistic framework introduced in [13–15]. The contributions of this thesis may be divided into two major parts. The first part is aimed at providing a thorough study and classification of the independence properties of multiresolution or multiscale (as they are called here) models, as well as providing a graph-theoretic framework by which to enumerate these independence properties. The second part of this thesis draws upon the ideas proposed in the first part, with the goal of developing efficient algorithms for solving the multiscale realization problem, *i.e.* designing multiscale models which exactly or approximately satisfy a given set of probabilistic constraints. As subsequent discussion reveals, the two threads of ideas – model independencies and model realization – are very much intertwined, and for this reason, these ideas are studied in conjunction with one another. Essentially, each chapter provides yet another perspective on the same general problem.

This thesis draws upon the large body of ideas generated from several decades worth of research in the area of multiresolution modeling, and in particular, it continues the lineage of research in the area of multiscale autoregressive (MAR) models introduced in [13–15]. The MAR framework has proven useful in a variety of applications including remote sensing [23, 30], geophysics [25, 70], oceanography [31,32], speech processing [61–63], and image processing [11,34,35,51,65,66,74,93,95]. One reason that these models have been so successfully applied in such diverse fields is due to the fact that the MAR framework is able to compactly and accurately model a wide variety of random phenomena. For example, MAR models have been shown to be well-suited for modeling one-dimensional Markov processes and some two-dimensional Markov random fields [73, 75], fractal-like processes [22, 24, 33, 74], and several other interesting examples [37, 38, 49, 52].

Another reason for the success of the MAR framework is that it offers an efficient set of tools for performing a number of important signal processing tasks. Among these are linear least-squares estimation [13, 15], likelihood calculation [76], calculation of error statistics [77], and sample path generation. In addition, this framework is well-suited for handling a number of challenges encountered in real-world signal processing problems such as the need to fuse data from multiple sources and modalities [14,23], handling large data sets [30], and dealing with irregularly-spaced and non-local measurements [21, 38]. Furthermore, the structure of these models allows computations to be performed in parallel – a feature which has become more relevant with the growing popularity of distributed networks [67].

During the same period of time in which the MAR framework has been developed, the related field of graphical models has grown in popularity – providing significant contributions to probabilistic modeling in general [58, 71, 84] and proving useful in a number of important application areas, perhaps most notably in the field of coding theory [6, 79, 87, 88]. A significant portion of the research performed in the area of graphical models seeks to find causal relationships in large datasets [85, 98], such as those encountered in the areas of medicine [53] and biology [96] for example. This area of study shares some connections with the MAR framework, but its objectives tend to be somewhat different. A second area of research in graphical models, and the one which most intersects with the MAR framework, has the goal of constructing accurate and parsimonious representations of random processes, so that estimation or inference tasks may be performed efficiently [57,80,101–103,107]. Because of this intersection, this thesis applies many of the important and relevant ideas developed within the graphical models community to the area of MAR models and to the richer class of multiscale models. This thesis also attempts to bridge the gap between these two bodies of research by incorporating ideas from each area, as well as their associated terminologies, into a cohesive framework for better understanding and analyzing the properties of multiscale models.

The remainder of this chapter provides a brief introduction to multiscale models as well as a description of the major problems to be addressed in this thesis. Subsequent chapters provide additional background material for the reader and as such, are meant to be self-contained. Nonetheless, each chapter draws upon results from previous chapters. For a detailed history of the MAR framework, see the introductory chapter of [38]; for a more thorough introduction to graphical models, see [59, 104]; and for a broad perspective of multiresolution models and their applications, see [109].

■ 1.1 Main Problems Addressed

The following sections provide a preview of the ideas presented in this thesis.

■ 1.1.1 Multiscale Models and Their Conditional Independencies

Multiscale models represent an important subclass of the more general class of directed acyclic models studied extensively in the graphical models literature [58, 71]. As we demonstrate, these types of models have a number of interesting properties and represent a generalization of the subclass of MAR models on which much of the probabilistic multiresolution modeling literature has focused up to this point [11,13,38,49,62]. Chapter 2 provides a formal definition of multiscale models, as well as a detailed study of their conditional independence or *Markov* properties. Chapter 3 then continues this study by introducing a graph-theoretic framework useful for enumerating these conditional independencies.

A multiscale model is a collection of random vectors $\{X_v\}$ which possesses a very special conditional independence structure. In particular, the vectors $\{X_v\}$ are indexed by the nodes or *vertices* of a tree (see for example the tree shown in Figure 1.1(a)), and any connection between vertices u and v (represented by an arrow) in the tree indicates a probabilistic constraint imposed on the vectors X_u and X_v. As we later discuss in more detail, the consequence of such probabilistic constraints is that the collection $\{X_v\}$ exhibits a particular set of conditional independencies called the *global Markov property* – a property which has been well-studied in the MAR literature [38,49].

As an example of the independence constraints imposed by the global Markov property, consider the tree shown in Figure 1.1(b). If vertex v is removed from this tree, the resulting graph consists of three distinct trees with vertices which lie in the sets A_v, B_v, and C_v shown in Figure 1.1(b). The global Markov property requires that random vectors X_{A_v}, X_{B_v}, and X_{C_v} be jointly independent

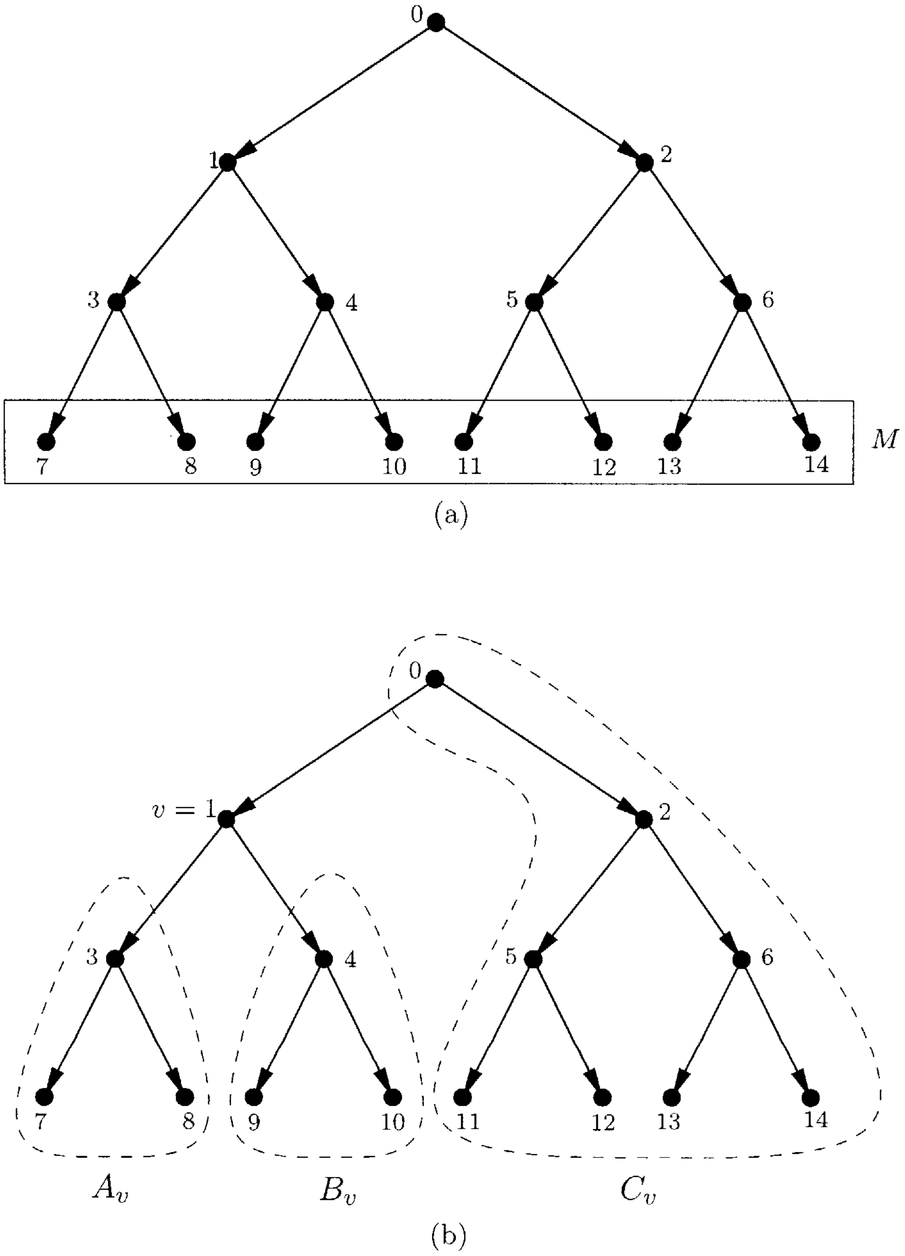


Figure 1.1. (a) An example of a tree containing vertices $0, 1, \ldots, 14$ which index the collection of random vectors $\{X_v\}$ associated with a multiscale model. (b) An example of the sets of vertices A_v, B_v, and C_v associated with the three different trees formed if vertex $v = 1$ is removed. In order for the global Markov property to be satisfied at vertex v, the random vectors indexed by these three sets must be jointly conditionally independent given X_v.

conditioned on random vector X_v.[1] Furthermore, the global Markov property is satisfied and the collection $\{X_v\}$ is a multiscale model if and only if a similar conditional independence constraint is satisfied for every vertex v in the tree.

While the global Markov property is useful for understanding the global independence structure of a multiscale model, the constraints imposed on $\{X_v\}$ by this property are numerous. For example, while we have described the constraints imposed on random vectors X_v, X_{A_v}, X_{B_v}, and X_{C_v} in Figure 1.1(b), similar constraints must be satisfied for every vertex v. Consequently, multiscale models must satisfy a large set of overlapping and competing constraints. In Chapter 2, we propose a less-complex set of constraints which we call the *reduced-order global Markov property*, and we show that this new type of Markovianity is equivalent to the global Markov property. As we discuss, this latter notion of Markovianity is not unique for a given tree; in fact, there are a number of equivalent reduced-order sets of constraints, each tied to a particular ordering of the vertices of a tree.

The benefit of the reduced-order global Markov property is that it helps to remove redundant independence constraints. As subsequent discussion reveals, redundant constraints are problematic for developing efficient multiscale realization algorithms. However, despite the simplification which the reduced-order global Markov property provides, the constraints which it imposes on $\{X_v\}$ are still severe in the sense that each random vector X_v must satisfy multiple competing constraints.

To address this issue, we introduce a novel form of Markovianity called the *marginalization-invariant Markov property*. This form of Markovianity is a function of a specified subset M of the vertices of a tree, and it imposes a constraint on the marginal density of the vectors $\{X_v\}_{v \in M}$ associated with a multiscale model. One interesting property of this form of Markovianity is that it is completely equivalent to the reduced-order global Markov property (and hence the global Markov property) when M contains all vertices of a tree. However, when M contains only a portion of the vertices, the conditional independence constraints imposed by this property are necessarily simpler than those required by the reduced-order global Markov property, and the degree of overlap between the constraints is less severe. Therefore, the marginalization-invariant Markov property represents a relaxation of the constraints imposed by the global Markov property, and the degree of relaxation is tied to the chosen set M.

In studying the marginalization-invariant Markov property, we take two different points-of-view. The first point-of-view, discussed in Chapter 2, is a set-theoretic perspective. This perspective states the constraints imposed by the marginalization-invariant Markov property in terms of a sequence of partitions of the vertices of a tree. As we demonstrate, these partitions can be obtained by a specific intersection of the constraints associated with the reduced-order global Markov property and the subset of vertices M. This further emphasizes the point that the marginalization-invariant Markov property is a relaxation of the reduced-order global Markov property and is strongly tied to the choice of M.

The second point-of-view, elaborated upon in Chapter 3, is a graph-theoretic perspective of the marginalization-invariant Markov property, and more generally, this point-of-view is based upon a novel theory which applies equally well to more complicated probabilistic models. In the first part of this chapter, we show that a rather simple statement about the factorization structure of a density $p(\cdot)$ provides a sufficient condition for the marginalization-invariant Markov property to be satisfied. The second part of this chapter then shows how this factorization constraint is equivalent to the set

[1] The notation X_A indicates the collection $\{X_v\}_{v \in A}$.

of conditional independencies required by the marginalization-invariant Markov property. To do this, we introduce two novel graph-theoretic constructs called *clique extensions* and *neighborhood separators*, and we use these constructs to demonstrate that the constraints of the marginalization-invariant Markov property can be ascertained by examining a special sequence of graphs.

Even though multiscale models represent a small subclass of the much broader class of graphical models, the exposition provided in this thesis demonstrates that they possess several interesting characteristics. The marginalization-invariant Markov property is the particular characteristic to which we devote most of our efforts. By itself, this property is interesting from a theoretical perspective, but it is not immediately obvious how it is useful for the multiscale realization problem. The following two sections provide a preview of why this Markov property is so important.

1.1.2 Realizing Exact Multiscale Models

The second major contribution of this thesis is in relating the conditional independencies possessed by multiscale models to the multiscale realization problem. The tie between these two ideas was first expressed in [49] while suggesting an interesting approach to multiscale realization based on *canonical-correlations analysis* [48], a method which provides one measure of conditional independence. Subsequently,the work of [38] suggests both a different measure of conditional independence termed *predictive efficiency* and a different sequential approach to the realization problem, termed *scale-recursive* realization.

The novelty of the ensuing discussion and our approach to the realization problem is that we separate the problem into two separate fundamental ideas: (1) the conditional independencies which a set of vectors $\{X_v\}$ must satisfy and (2) how to measure the degree to which these conditional independencies are satisfied. As we show, an answer to the first idea is given by the constraints imposed by the marginalization-invariant Markov property. The second idea can be attacked from several equally viable perspectives depending on the particular problem at hand. For example, the work of [49] focuses on canonical correlations, while the work of [38] focuses on predictive efficiency. Our approach to this second idea focuses on measuring conditional independence using the *Kullback-Leibler divergence* [17].

An additional benefit of the dichotomy between conditional independence constraints and how to measure them is that we can focus our attention on these two different ideas in a systematic manner. In Chapter 2 and the first part of Chapter 3, we solely focus on the exact multiscale realization problem, and we show that it may be solved by satisfying the constraints imposed by the marginalization-invariant Markov property. Then, in the last part of Chapter 3 and in Chapter 4, we focus on the impact of only approximately satisfying the marginalization-invariant constraints, and we show how such an approximation affects the overall accuracy of the multiscale model.

As a brief introduction to these ideas, the multiscale realization problem consists of three main ingredients: a set of vectors $\{X_v\}$ indexed by the vertices of a tree, a set M composed of a subset of the vertices of a tree, and a specified marginal density $p(x_M)$. In the exact realization problem, the vectors $\{X_v\}_{v \in M}$ are constrained to have the density $p(x_M)$, while the remaining vectors $\{X_v\}_{v \notin M}$ are so-called design vectors and are unconstrained. The goal is to specify these design vectors so that the complete collection $\{X_v\}$ corresponds to a multiscale model. More specifically, the correct specification of the design vectors $\{X_v\}_{v \notin M}$ requires finding a conditional density $p(x_{v \notin M} | x_M)$ so that the density $p(x_{v \notin M} | x_M) p(x_M)$ has the factorization structure required for a multiscale model.

As an example, consider a zero-mean Gaussian density $p(x_M)$ with a covariance matrix whose

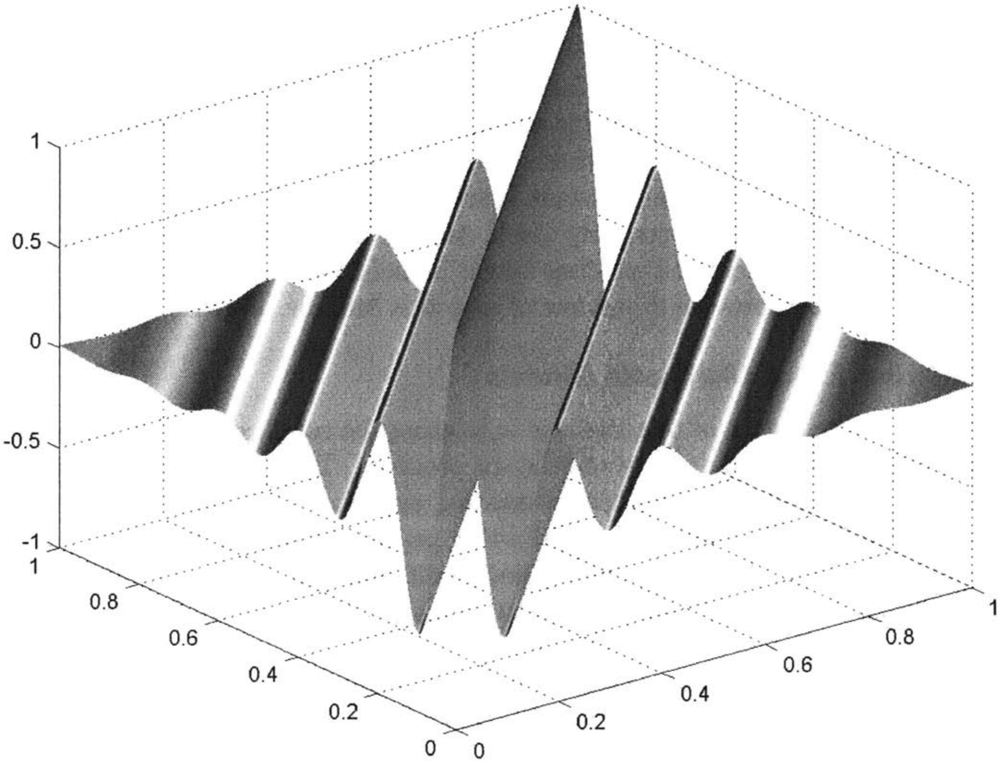


Figure 1.2. Surface plot of the entries of a 512 × 512 covariance matrix which corresponds to a damped sinusoid.

entries are graphically depicted by the height of the surface shown in Figure 1.2 and where M corresponds to the terminal vertices of the tree previously shown in Figure 1.1(a). In this example, the multiscale realization problem consists of identifying random vectors $X_0, X_1, \ldots, X_6$, *i.e.* the vectors indexed by the non-terminal vertices of the tree, so that the marginal constraint $p(x_M)$ is satisfied and so that the collection $\{X_v\}$ corresponds to a multiscale model. In choosing these design vectors, we must simultaneously satisfy the tradeoff between the marginal constraint $p(x_M)$ and the constraints imposed by the global Markov property.

The benefit of describing the multiscale realization problem in such general terms is that the density $p(x_M)$ can represent any type of probabilistic constraint. Specifically, with this view of the realization problem, we can handle the following important specifications of the process X_M:

(1) An exact set of probabilistic constraints represented by a complete density $p(x_M)$.

(2) A partial set of probabilistic constraints represented by a subset of the marginals of $p(x_M)$.

(3) A set of realizations (observations) of the process X_M.

The second type of specification is encountered in the covariance completion problem where only a partial set of second-order statistics is known for a given process (see [36, 38] for a discussion of this type of specification). The third type of specification is encountered when dealing with data, where the density $p(x_M)$ corresponds to the empirical density associated with a given set of observations. In this thesis, we focus only on a complete characterization of the density $p(x_M)$, but the theoretical foundation derived here applies equally well to cases (2) and (3).

The novelty of our approach to the multiscale realization problem is derived from the fact that it is based on the marginalization-invariant Markov property. By definition, the constraints imposed by this property are precisely the constraints which a set of vectors $\{X_v\}$ must satisfy in order for the realization problem to have an exact solution. When the set M contains only terminal vertices of a tree, we demonstrate that these constraints may be ordered in such a way that permits a sequential approach to solving the realization problem. In cases where M includes non-terminal vertices of a tree,[2] we propose an alternative realization problem which introduces additional design variables. This problem is a generalization of the state augmentation approach proposed in [21]. Using these additional design variables, as well as a generalized form of the marginalization-invariant Markov property, we devise a sequential realization procedure for these types of problems as well.

■ 1.1.3 Realizing Approximate Multiscale Models

For any choice of density $p(x_M)$ where M corresponds to the terminal vertices of a tree, the multiscale realization problem always has an exact solution. In particular, the dimensions of the design vectors X_v, $v \notin M$, may be increased to the point where an exact, perhaps trivial, solution exists. This approach to the realization problem is undesirable, however, because the value of a multiscale model is derived from the fact that it provides a simple factored representation of a density $p(x_M)$. If the dimensionality of this representation is significantly larger than that of X_M, then a multiscale model provides little or no benefit for estimation and other signal processing tasks.

To address this issue, the latter part of Chapter 3 proposes a relaxed version of the exact multiscale realization problem where the density associated with random vectors $\{X_v\}_{v \in M}$ is only required to approximately match the desired marginal $p(x_M)$. Specifically, we constrain the dimensions of the design vectors $\{X_v\}_{v \notin M}$, and we identify the choices for these vectors which generate a collection $\{X_v\}$ that "best" approximates the marginal density $p(x_M)$. Of course, the definition of best depends on the chosen criterion. One natural criterion, and the one we choose, is the Kullback-Leibler divergence.

Even though the Kullback-Leibler divergence is not a metric in the strict sense, it provides some notion of the deviation between two probability densities, and it has several nice properties which we use to our advantage. Using this particular notion of deviation, we propose a cost function for the approximate realization problem which can be decomposed into a sum of different terms. We show that each term represents the degree to which the constraints of the marginalization-invariant Markov property are satisfied. Therefore, even when considering a relaxed version of the multiscale realization problem, we can show that the severity of the approximation is directly tied to how well the marginalization-invariant Markov property is satisfied, and we can directly measure the degree of approximation.

In Chapter 4, we propose an iterative procedure for solving the approximate multiscale realization problem. This type of iterative approach to realization represents a significant departure from the sequential procedures previously proposed for MAR models [38, 49], and in fact, it has

[2] This scenario is encountered in applications where both lower- and high-resolution data must be fused into a single cohesive probabilistic model. For details, see the groundwater problem studied in [21, 25].

more commonality with the well-known EM algorithm [27] used for realizing more general graphical models [59]. As we show, our proposed iterative procedure is identical to EM when specific parameterized realization problems are considered.

The novelty of our discussion of EM lies in the fact that we can provide a direct link between the EM algorithm and the conditional independencies possessed by multiscale models. As we show, the two steps of the EM algorithm seek a tradeoff between the two types of constraints involved in the approximate realization problem. Specifically, a multiscale model must (1) provide an accurate approximation to a density $p(x_M)$ and (2) approximately satisfy the constraints imposed by the global Markov property. We prove that the EM algorithm tries to find an optimal solution by a two-step process that alternates between optimizing constraints (1) and (2).

Another contribution provided in Chapter 4 is a specific algorithm for solving the Gaussian multiscale realization problem using EM.[3] While EM has been used for realizing multiscale models from data [64], it has not been used to generate a model which approximates the second-order statistics of a process. This particular problem has been considered in [49] and [38] but not from the perspective of EM. In the final part of Chapter 4, we derive an efficient method for calculating the matrix quantities necessary to perform the EM iterations, and we provide several illustrative examples of the performance of this algorithm in practice.

1.2 Thesis Organization

The remainder of this thesis is organized in the following manner.

Chapter 2 – Multiscale Models and Markovianity

This chapter formally defines the class of multiscale models considered in this thesis and studies their independence properties. It is well-known that multiscale models satisfy a so-called global Markov property – essentially a list of conditional independence statements. As we demonstrate, this list contains a significant number of redundancies, and we suggest a set-theoretic method for generating a less-complex but equivalent set of conditional independencies. In addition, we introduce a novel notion of Markovianity called marginalization-invariant Markovianity which ties the list of independencies to a given marginal constraint.

This chapter also formally defines the multiscale realization problem considered in this thesis. We then use the marginalization-invariant Markov property to suggest a sequential procedure for constructing multiscale models which satisfy a specified marginal constraint. In cases where such a sequential procedure is not possible, we introduce an alternative realization procedure called state augmentation, and we show that every type of multiscale realization problem can in theory be solved using this latter procedure.

Chapter 3 – Realizing Multiscale Models: A Graph-Theoretic Perspective

This chapter continues to investigate the ideas introduced in Chapter 2 by presenting a graph-theoretic framework for analyzing the conditional independencies associated with multiscale models. This framework is useful because it provides a simple method for generating a list of conditional independencies by examining a special sequence of graphs. In this discussion, we introduce two new

[3] The Gaussian realization problem requires the density $p(x_M)$ to be Gaussian and requires the vectors $\{X_v\}$ of the multiscale model to be jointly Gaussian.

graph-theoretic constructs, and we provide a rather general result concerning the Markov properties of a broad class of graphical models. Using this general result, we then prove two important special cases which were stated in Chapter 2 but not proven.

In this chapter, we also introduce a relaxed version of the multiscale realization problem which we formulate as a constrained optimization problem. Using the Kullback-Leibler divergence as a measure of approximation, we state necessary conditions for optimality. In addition, we demonstrate that the approximate realization problem may be solved by minimizing a special upper bound, and we show that this upper bound may be decomposed into a sum of terms, where each term directly measures the degree to which the marginalization-invariant Markov property is satisfied.

Chapter 4 – Realizing Approximate Multiscale Models Using EM

This chapter continues the study of the approximate multiscale realization problem by proposing an iterative procedure for solving it. Using the theory developed in Chapter 3, we show that the proposed procedure seeks to find an appropriate tradeoff between satisfying a specified marginal constraint and satisfying the global Markov property. From a slightly different perspective, we also show how this procedure seeks to minimize an upper bound – a bound which is tight for exact solutions to the realization problem.

Up until the second half of Chapter 4, the entire class of multiscale models is considered in its totality, and all results are stated generically for the class as a whole. In the second half of this chapter, we focus on developing specialized realization algorithms for a parameterized subclass of multiscale models, and we develop a parameterized version of the iterative procedure proposed in the first half of the chapter. We subsequently prove that this latter procedure is equivalent to the EM algorithm, and using this procedure, we derive an efficient algorithm for finding local optima of the approximate Gaussian multiscale realization problem.

Chapter 5 – Conclusions and Future Research Directions

This chapter provides concluding remarks and summarizes the main contributions of this thesis. Several extensions to the ideas presented here are discussed, with a number of suggestions for further research.

Chapter 2

Multiscale Models and Markovianity

THE primary purpose of this chapter is to introduce the class of multiscale models and to discuss their so-called *Markov* properties. Through this exploration, we demonstrate that understanding these Markov properties can be advantageous in the development of efficient multiscale realization algorithms. The results of this chapter also establish a common framework from which to view several important results in multiscale realization theory [21, 38, 49, 73].

In order to achieve these goals, it is necessary to introduce basic results from graph theory and provide some notation in Sections 2.1 and 2.2. In Section 2.3, the multiscale model is formally defined, and the important subclasses of Gaussian multiscale models and internal multiscale models are introduced. Sections 2.4, 2.5, and 2.6 introduce and discuss the Markov properties of multiscale models. Section 2.7 formally defines the multiscale realization problem and shows how a thorough understanding of the Markov properties of multiscale models is an important step in better understanding the issues involved in the realization problem. Section 2.8 provides several examples to illustrate the relationship between the framework established in this chapter and earlier work in this area.

2.1 Some Graph Theory

A graph $\mathcal{G} = (V, E)$ is an ordered pair of sets[1], where V is called the *vertex set* and $E \subset V \times V$ is called the *edge set*. We henceforth assume that graphs are finite so that $|V| < \infty$, and we impose the additional restriction that for all $v \in V$, the edge $(v, v) \notin E$. Consequently, graphs contain no self-loops. The edge set E may contain two types of edges: *directed* and *undirected*. An edge is directed if $(u, v) \in E$ and $(v, u) \notin E$, while an edge is undirected if $(u, v) \in E$ and $(v, u) \in E$. When an edge is undirected, we use the shorthand $\{u, v\} \in E$ to denote both $(u, v) \in E$ and $(v, u) \in E$.

So far, a graph has been described as a purely mathematical object, but it is also convenient to consider it a "graphical" object. To do this, we represent a vertex by a small dot, an undirected edge by a line connecting two vertices, and a directed edge (u, v) by an arrow pointing from vertex u to vertex v. Figure 2.1(a) provides such an illustration of a graph, where in this case the vertex set V consists of positive integers. A graph containing only undirected edges is called an *undirected graph* as shown in Figure 2.1(b), and a graph containing only directed edges is called a *directed graph* as shown in Figure 2.1(c).

Given two graphs $\mathcal{G} = (V, E)$ and $\mathcal{G}' = (V', E')$ with $V \subseteq V'$ and $E \subseteq E'$, $\mathcal{G}$ is called a *subgraph* of $\mathcal{G}'$, and $\mathcal{G}'$ is called a *supergraph* of $\mathcal{G}$. The subgraph of $\mathcal{G}$ *induced* by a set of vertices $U \subseteq V$ and denoted by $\mathcal{G}(U)$ is the graph with vertices U and edges $E \cap (U \times U)$. This means

[1] We use upper case English letters to denote sets.

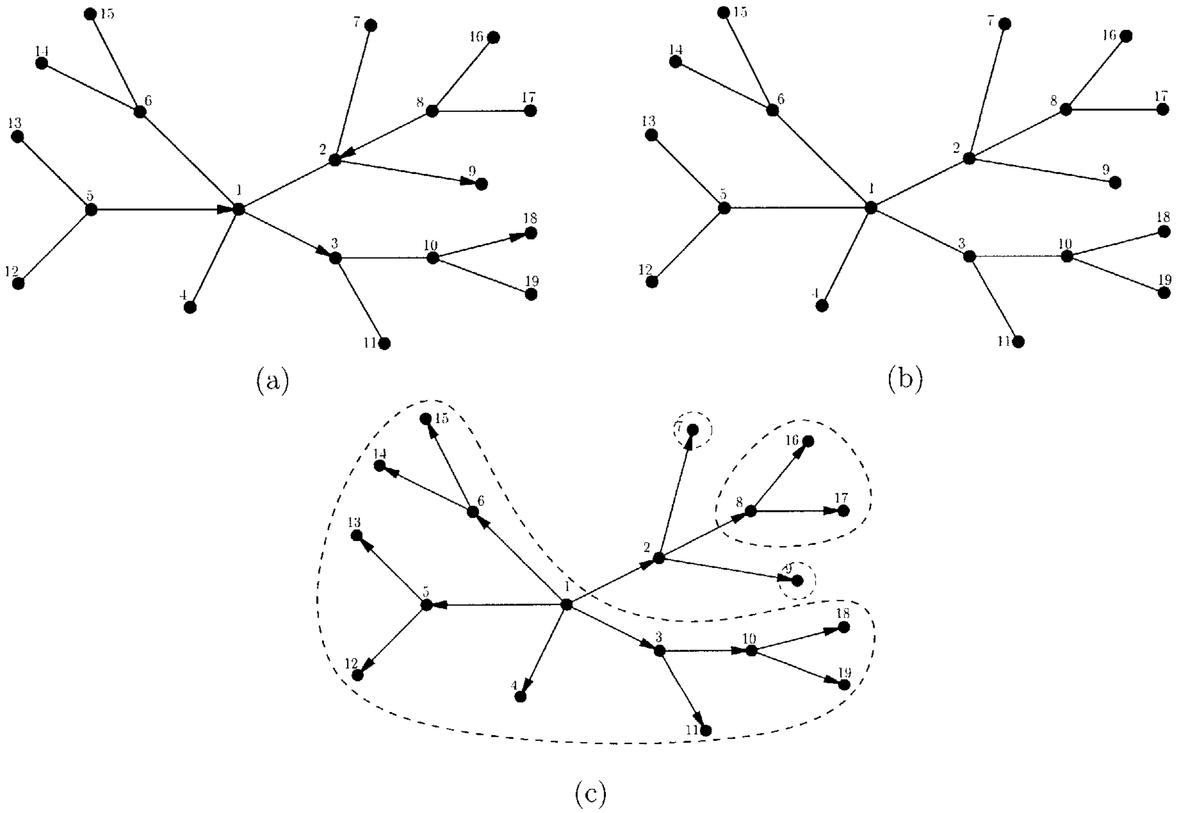


Figure 2.1. (a) Graph with directed and undirected edges. (b) Undirected graph and a tree. (c) Directed graph and a rooted tree. The dashed lines show the four subtrees formed by removing vertex 2.

that the induced subgraph only contains edges $(u, v) \in E$ with both $u, v \in U$. For example, in Figure 2.1(c), the subgraph induced by the vertices $U = \{1, 5, 12, 13\}$ contains the directed edges $\{(1,5),(5,12),(5,13)\}$.

If (u, v) is a directed edge in a graph $\mathcal{G} = (V, E)$, then v is called the *child* of u, and u is called the *parent* of v. We introduce two special functions π and χ which map elements $v \in V$ into subsets of V as follows,

$$\pi(v) = \{u \in V | (u, v) \text{ is a directed edge in } E\} \tag{2.1a}$$

$$\chi(v) = \{u \in V | (v, u) \text{ is a directed edge in } E\}. \tag{2.1b}$$

Thus, $\pi(v)$ and $\chi(v)$ respectively contain the parents and children of vertex v. The *closure* of a subset of vertices $U \subseteq V$ with respect to a directed graph $\mathcal{G}$ contains U as well as the parents of all elements in U and is defined as follows,

$$\bar{U} \triangleq (\cup_{u \in U} \pi(u)) \cup U. \tag{2.2}$$

The closure of $U = \{10, 18, 19\}$ in Figure 2.1(c) is $\bar{U} = \{3, 10, 18, 19\}$.

A *walk* of length n from vertex v_0 to vertex v_n (denoted $[v_0, \ldots, v_n]$) in a graph $\mathcal{G}$ is a sequence of vertices $v_0, \ldots, v_n$ such that (v_{i-1}, v_i) is an edge in the graph for $1 \leq i \leq n$. Similarly, a *path* is a walk $[v_0, \ldots, v_n]$ such that the vertices $v_0, \ldots, v_n$ are distinct. If there is a path between every two distinct vertices $u, v \in V$, then we say that the graph is *connected.* For example, the graph in Figure 2.1(b) is connected. A *cycle* of length n is a path of length n but with the modification that $v_0 = v_n$. Given two vertices v and v' such that there exists a path from v to v' but no path from v' to v, we call v an *ancestor* of v', and we call v' a *descendent* of v. In Figure 2.1(c), vertex 1 is an ancestor of vertex 18, while vertex 18 is a descendent of vertex 1.

A *tree* is an undirected graph which is connected and contains no cycles. Figure 2.1(b) shows one example of a tree. A *rooted tree* is constructed from a tree by choosing a vertex called the *root* and then replacing all undirected edges with directed edges which point away from the root. Figure 2.1(c) shows the rooted tree obtained from the tree in Figure 2.1(b), by choosing vertex 1 as the root. In most of the discussion to follow, we focus on graphs which are rooted trees, and we devote the next section to further discussing these types of graphs.

2.2 Rooted Trees and the Notion of Scale

Suppose the graph $\mathcal{G}_{\preceq} = (V, E)$ is a rooted tree.[2] We use the notation $\mathcal{G}_{\preceq}$ because as we discuss here, a rooted tree has a natural partial order $\preceq$. Suppose v_0 is the root of $\mathcal{G}_{\preceq}$, and consider the following function which maps vertices into the nonnegative integers,

$$m : V \longrightarrow \mathbb{N}, \quad m(v) = \text{number of edges between } v_0 \text{ and } v, \tag{2.3}$$

and where we define $m(v_0) \triangleq 0$. In Figure 2.1(c), for example, $m(v) = 1$ for $v = 2, \ldots, 6$ and $m(v) = 2$ for $v = 7, \ldots, 15$.

We refer to the value $m(v)$ as the *scale* of vertex v, e.g. the scale of v_0 is 0. In subsequent sections, we map random vectors to each of the vertices of a graph, and at that point, this particular word choice becomes more meaningful, since it says something about the resolution or scale of the

[2] For simplicity, we will occasionally refer to $\mathcal{G}_{\preceq}$ as a tree, understanding that it is really a rooted tree.

process mapped to the graph. For our purposes, though, it is notationally simpler to associate scale with the vertices rather than the corresponding random process.

For a rooted tree, there exists a natural partial order $\preceq$ on the vertices V given by

$$v \preceq v' \quad \text{if } v' \text{ is a descendent of } v. \tag{2.4}$$

For any partial order, we say that two elements are *comparable* if either $v \preceq v'$ or $v' \preceq v$; otherwise, they are *incomparable*. For the partial order $\preceq$, two vertices are comparable only when one is a descendent of the other. A subset of a partially ordered set is called a *chain* if all elements are comparable. The maximal (with respect to inclusion) chains of $\preceq$ are subsets $\{v_0, \ldots, v_n\}$ of V where $[v_0, \ldots, v_n]$ is a path in the graph $\mathcal{G}_{\preceq}$ starting at the root v_0 and ending at v_n, a terminal vertex with no children – called a *leaf* vertex. The leaves of the tree in Figure 2.1(c) are 4,7,9, and 11–19. It is convenient to define the following subsets of leaf vertices,

$$L_v \triangleq \{u \in V | u \succeq v, \chi(u) = \emptyset\}, \tag{2.5}$$

i.e. L_v contains the leaf vertices which descend from vertex v. Of course, L_{v_0} contains all leaf vertices, and if v is itself a leaf vertex then $L_v = \{v\}$. In Figure 2.1(c), $L_3 = \{11, 18, 19\}$.

We also define several special subgraphs which we later use. A *subtree* of a rooted tree $\mathcal{G}_{\preceq}$ is a subgraph of $\mathcal{G}_{\preceq}$ which is itself a rooted tree with root v (not necessarily equal to v_0). One important subtree of $\mathcal{G}_{\preceq}$ is the subgraph induced by the vertices

$$S_v \triangleq \{u \in V | v \preceq u\}, \tag{2.6}$$

i.e. the set containing v and all vertices which are descendants of v. For example, S_3 is equal to $\{3, 10, 11, 18, 19\}$ for the rooted tree shown in Figure 2.1(c). We introduce the notation $S_v^c \triangleq V - S_v$ to indicate the vertices V not contained in S_v. Note that the subgraph induced by S_v^c is a subtree with root v_0 which does not contain the vertices S_v.

Suppose that a vertex v has q children $\chi(v)$, and imagine removing the vertex v from the graph $\mathcal{G}_{\preceq}$, then the resulting graph consists of $q+1$ subtrees induced by the sets S_v^c and S_u for $u \in \chi(v)$. Henceforth, we say that vertex v *separates* the graph into $q+1$ subtrees. For example, the dashed lines in Figure 2.1(c) indicate the subtrees separated by vertex 2. Also, recall that the closure of a set of vertices includes all parent and neighboring vertices of the set. For all $v \neq v_0$, the closure of S_v is simply $\bar{S}_v = S_v \cup \{\pi(v)\}$.

■ 2.3 Multiscale Models

Given a graph $\mathcal{G} = (V, E)$, suppose we map to each vertex $v \in V$ a random vector X_v, taking values in some space $\mathcal{X}_v$. In so doing, we have created a random process indexed by a graph $\mathcal{G}$, and we henceforth use the ordered pair $(X, \mathcal{G})$ to represent such a process, where $X = \{X_v\}_{v \in V}$ is the set of random vectors mapped to the vertices of $\mathcal{G}$. When the graph is a rooted tree $\mathcal{G}_{\preceq}$, we say that $(X, \mathcal{G}_{\preceq})$ is a *tree-indexed process*. Given a tree-indexed process, we call the process indexed by the leaves of the tree, *i.e.* $X_{L_{v_0}}$, the *finest-scale process*, and if $m(v) < m(u)$, we say that X_v is of *coarser-scale* than X_u.[3]

[3] The reason for this particular terminology is that if a tree-indexed process $(X, \mathcal{G}_{\preceq})$ represents multiresolution data, then the leaf vertices contain the finest-resolution data, while vertices closer to the root represent a coarsening of the data.

By itself, a tree-indexed process is rather uninteresting. However, if we impose additional requirements on the random variables mapped to the graph – specifically, if we require the probability density to factor in a certain way – then such a process displays very interesting conditional independence properties. Much of the remaining discussion in this chapter is devoted to further understanding this idea. In this section, we introduce the type of factorization that we consider, and in subsequent sections, we discuss the conditional independence properties that are a consequence of this factorization.

2.3.1 Definition of Multiscale Models

Before defining the class of multiscale models, it is useful to provide one general definition associated with directed acyclic graphical models. Note that given a subset of vertices $U \subset V$, we use the notation X_U to represent the collection of random vectors $\{X_u\}_{u \in U}$.

Definition 2.1 (Recursive Factorization).
Let $(X, \mathcal{G})$ be a process indexed by a directed acyclic graph $\mathcal{G}$, where X has a probability density p. Then, p admits a *recursive factorization* with respect to $\mathcal{G}$ if there exist conditional probability densities $p(x_v|x_{\pi(v)})$ such that

$$p(x) = \prod_{v \in V} p(x_v|x_{\pi(v)}), \tag{2.7}$$

where $p(x_v|x_{\pi(v)}) \triangleq p(x_v)$ when $\pi(v)$ is empty. ◀

Therefore, a recursive factorization with respect to a directed acyclic graph $\mathcal{G}$ allows a density p to be factored as a product of so-called "local" probability densities each of which involve only the random vectors X_v and $X_{\pi(v)}$.[4] Figure 2.2(a) shows one example of a recursive factorization. Notice that we have chosen to represent the vertices in Figure 2.2(a) by large shaded circles rather than the smaller dots used in earlier graphs; this is simply to indicate that vectors $\{X_v\}$ have been mapped to the vertices of the graph and that the process recursively factors according to the graph.

Definition 2.2 (Multiscale Model).
A *multiscale model* is a tree-indexed process $(X, \mathcal{G}_{\preceq})$ with a density p that admits a recursive factorization with respect to $\mathcal{G}_{\preceq}$. ◀

Examining Definitions 2.1 and 2.2, it is clear that multiscale models are a subclass of the larger set of models whose densities admit a recursive factorization with respect to directed acyclic graphs. This broader class of models, commonly termed *directed graphical models*, has been studied in detail (see [71] for example), but as we show, there are still open and interesting questions for the smaller class of multiscale models. In addition, as later examples illustrate, multiscale models are a rich and powerful set of models, providing utility in a variety of applications.

Note that since a multiscale model is defined on a rooted tree, the parent set $\pi(v)$ is a singleton for each non-root vertex $v \in V$, and consequently, each density $p(x_v|x_{\pi(v)})$ can be associated with an edge of the graph $\mathcal{G}_{\preceq}$. Figures 2.2(b), (c), and (d) show three examples of multiscale models defined on different tree structures. The multiscale model in Figure 2.2(b) is defined on a graph

[4] Notice that Definition 2.1 assumes that X permits a density p. We shall continue to assume that this condition holds for all random vectors, but for clarity of exposition, we will no longer remind the reader of this fact.

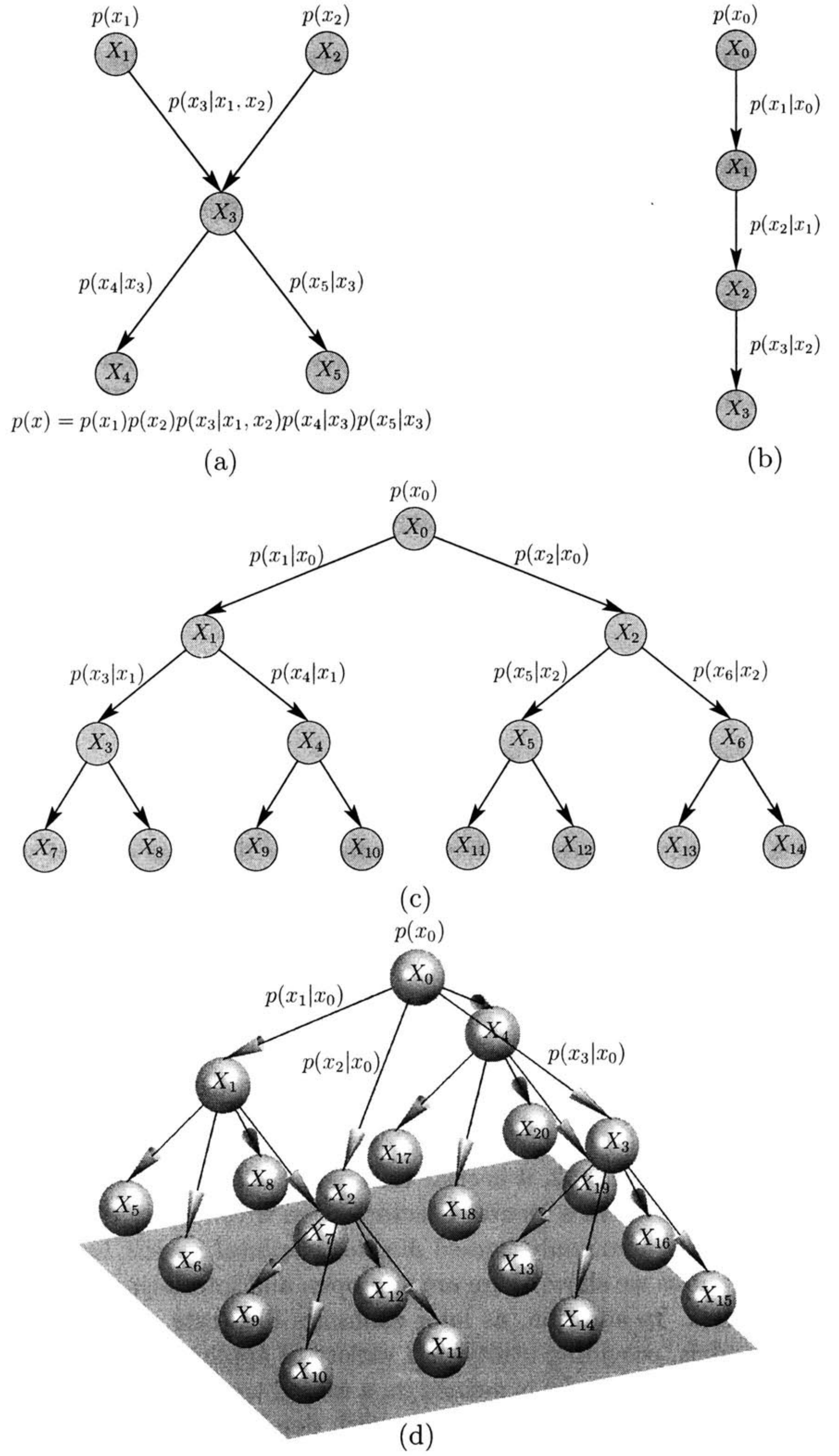


Figure 2.2. (a) A model with a density that recursively factors according to a directed acyclic graph which is not a rooted tree. (b–d) Three multiscale models defined on different graph structures: (b) A monadic tree. (c) A dyadic tree. (d) A quad tree.

where each non-leaf vertex has a single child, *i.e.* a monadic tree. In the time-series literature, a model defined on a monadic tree is more commonly called a *Markov chain*, but since this type of model is a special case of Definition 2.2, we shall (where appropriate) consider it a multiscale model.

Figures 2.2(c) and (d) show two other possible graph structures, namely a *dyadic* tree and a *quad* tree. Notice that we have drawn the quad tree in Figure 2.2(d) as a three-dimensional object; this is to emphasize the fact that such a structure can be used to model a two-dimensional random field. However, the structure of this graph is no different than the planar graph shown in Figure 2.2(c), except that each non-leaf vertex now has four children rather than two. Examining Figures 2.2(b),(c), and (d), notice that a multiscale density $p(x)$ is completely specified by a root density $p(x_0)$ and the "local" interactions $p(x_1|x_0)$, $p(x_2|x_0)$, ... We explore this idea further in the next section for a particular subclass of multiscale models.

■ 2.3.2 Gaussian Multiscale Models

Gaussian multiscale models are an important subclass of multiscale models where each density $p(x_v|x_{\pi(v)})$ is normally distributed. For our purposes, this is an important subclass since it is the focus of the model realization procedures described in subsequent chapters. It is also the predominant type of multiscale model investigated up to this point [13, 14, 21, 29, 30, 38–40, 47, 49, 50, 73, 75–77].

For simplicity, assume that the random vectors $\{X_v\}$ are all zero-mean, and denote the covariance of X_v by P_{x_v} and the cross-covariance between X_v and $X_{\pi(v)}$ by $P_{x_v x_{\pi(v)}}$. Since $p(x_v|x_{\pi(v)})$ is a normal density, it is characterized by its mean μ_v and covariance $\widetilde{P}_v$[5]; it is well-known that these two quantities are given by[6]

$$\mu_v = P_{x_v x_{\pi(v)}} P_{x_{\pi(v)}}^{-1} x_{\pi(v)} \tag{2.8a}$$

$$\widetilde{P}_v = P_{x_v} - P_{x_v x_{\pi(v)}} P_{x_{\pi(v)}}^{-1} P_{x_v x_{\pi(v)}}^T. \tag{2.8b}$$

As (2.8) indicates, this particular class of multiscale models is completely parameterized by the set of covariances $\mathcal{P}_{\mathcal{G}_{\preceq}} \triangleq \{P_{x_{v_0}}\} \cup \{P_{x_v}, P_{x_v x_{\pi(v)}}\}_{v \in V - \{v_0\}}$, where v_0 is the root vertex.

Consider also a different set of parameters A_v, Q_v defined as follows for all $v \neq v_0$,

$$A_v \triangleq P_{x_v x_{\pi(v)}} P_{x_{\pi(v)}}^{-1} \tag{2.9a}$$

$$Q_v \triangleq \widetilde{P}_v = P_{x_v} - P_{x_v x_{\pi(v)}} P_{x_{\pi(v)}}^{-1} P_{x_v x_{\pi(v)}}^T. \tag{2.9b}$$

Note that (2.9) defines a mapping from the covariances $\mathcal{P}_{\mathcal{G}_{\preceq}}$ to the parameters $\{A_v, Q_v\}_{v \in V - \{v_0\}}$. This mapping is in fact invertible given the root covariance $P_{x_{v_0}}$ since the remaining covariances can be recursively computed along all chains of the partial order $\preceq$ as follows,

$$P_{x_v x_{\pi(v)}} = A_v P_{x_{\pi(v)}}$$

$$P_{x_v} = Q_v + A_v P_{x_v x_{\pi(v)}}^T.$$

[5] We henceforth use the notation $X \sim \mathcal{N}(\mu, P)$ to denote a random vector X which is normally distributed with mean μ and covariance P.

[6] For simplicity, we assume that all covariances $P_{x_{\pi(v)}}$ are positive definite. This is a reasonable assumption, since in most cases of practical interest, a model with singular covariances $P_{x_{\pi(v)}}$ may be replaced by a model with non-singular covariances, without degrading the model fidelity.

Consequently, an equivalent parametrization of this class of multiscale models is given by $\{P_{x_{v_0}}\} \cup \{A_v, Q_v\}_{v \in V - \{v_0\}}$. One reason for considering this particular parametrization is that random draws from the conditional distribution $p(x_v|x_{\pi(v)})$ may be generated via the simple recursive equation,

$$x_v = A_v x_{\pi(v)} + \widetilde{x}_v, \tag{2.11}$$

where $\widetilde{x}_v$ is drawn from a zero-mean Gaussian distribution with covariance Q_v. This suggests one benefit of multiscale models; namely, assuming that $p(x)$ admits a recursive factorization, then samples from $p(x)$ may be easily generated using (2.11).

Besides the benefit of efficient simulation, these types of multiscale models are efficient in at least two other important ways. First, if X is a zero-mean Gaussian random vector with covariance P_x and a density satisfying $p(x) = \prod_{v \in V} p(x_v|x_{\pi(v)})$, then there must exist an invertible mapping between the covariances $\mathcal{P}_{\mathcal{G}_{\preceq}}$ and P_x. For problems where the dimension of P_x is large, storing only the covariances $\mathcal{P}_{\mathcal{G}_{\preceq}}$ can be a significant improvement over storing all of P_x.

Second, Gaussian multiscale models admit an efficient inference or estimation algorithm. Suppose that at each vertex $v \in V$, there exist observations of the form

$$Y_v = C_v X_v + W_v, \tag{2.12}$$

where $W_v \sim \mathcal{N}(0, R_v)$ and is uncorrelated with X_v. Notice that (2.12) in combination with (2.11) is reminiscent of the state-space representation used in time-series models. For time-series, it is well-known that it is possible to calculate the conditional distribution $p(x_v|\{y_v\}_{v \in V})$ in a recursive manner (with respect to time) via the Kalman filter [60] and Rauch-Tung-Striebel smoother [86]. The generalization to multiscale models has also been shown to be possible [13, 14], where in this case, the recursion occurs with respect to scale. If we let d_v represent the dimension of each vector X_v, the complexity of the inference algorithm has been shown to be proportional to $\sum_{v \in V} d_v^3$. The reason that this complexity is possible is that $p(x)$ can be parameterized in terms of local rather than global parameters and because these types of models obey the *global Markov property* discussed in Section 2.4.

■ 2.3.3 Internal Multiscale Models

Another important type of multiscale model is the subclass of so-called *internal* multiscale models. The notion of an internal model is originally derived from the time-series literature [72], but this idea has been successfully applied to the more general class of multiscale models and studied extensively in the context of the multiscale realization problem [21, 24, 38–40, 49, 50]. This section provides a few generalizations of the ideas originally presented in [38–40].

We begin with the definition of an internal tree-indexed process.

Definition 2.3 (Internal Tree-Indexed Processes).
A tree-indexed process $(X, \mathcal{G}_{\preceq})$ is *internal* if for all non-leaf vertices v, X_v is a deterministic function of the process indexed by the leaf vertices descending from v, *i.e.* for each vertex v, there exists some function $f_v(\cdot)$ such that

$$X_v = f_v(X_{L_v}). \tag{2.13}$$

◀

Notice that the process mapped to the leaf vertices, *i.e.* $X_{L_{v_0}}$, and the collection of functions $\{f_v(\cdot)\}_{v \in V - L_{v_0}}$ completely specify an internal tree-indexed process. In addition, the process $X_{L_{v_0}}$ contains all of the inherent randomness in this type of tree-indexed process, since $X_{V-L_{v_0}}$ is a deterministic function of $X_{L_{v_0}}$. Consequently, the complete density $p(x_V)$ is degenerate for an internal tree-indexed process.

Given the above definition, an *internal multiscale model* is an internal tree-indexed process $(X, \mathcal{G}_{\preceq})$ with a density that also recursively factors according to $\mathcal{G}_{\preceq}$. As one might suspect, requiring a model to be both internal and have a recursive factorization leads to interesting properties of (and constraints on) the density $p(\cdot)$. To investigate these further, we introduce the notion of locally internal tree-indexed processes.

Definition 2.4 (Locally Internal Tree-Indexed Processes).
A tree-indexed process $(X, \mathcal{G}_{\preceq})$ is *locally internal* if for all non-leaf vertices v, X_v is a deterministic function of the process indexed by the child vertices of v, *i.e.* for each vertex v, there exists some function g_v such that

$$X_v = g_v(X_{\chi(v)}). \tag{2.14}$$

◀

Notice that the class of locally internal tree-indexed processes includes the class of internal tree-indexed processes. This is because we can always recursively compose the local parent-child functions in (2.14) to write X_v as a function of the process X_{L_v}. The reverse operation is however not true in general – an internal tree-indexed process may not be a locally tree-indexed process, but these two types of processes are equivalent if an additional constraint is imposed, namely that the process is in fact a multiscale model.

Proposition 2.1 (Equivalence of Internal and Locally Internal Multiscale Models).
A multiscale model is internal if and only if it is locally internal.

Proof. See Appendix A.1. ■

This is a valuable result because it shows that all internal multiscale models can be parameterized by a set of "localized" functions $\{g_v(\cdot)\}$; that is, for each non-leaf vertex v, $X_v = g_v(X_{\chi(v)})$ is only a function of $X_{\chi(v)}$. The proof of this result relies entirely on the conditional independence properties associated with the recursive factorization, which we now examine in detail.

■ 2.4 The Global Markov Property

In this section, we characterize the conditional independence or *Markov* properties of multiscale models, and we show that these Markov properties can be discerned from the underlying graph $\mathcal{G}_{\preceq}$. To provide intuition, we begin with an example. This example addresses three important questions that underly the remainder of the results in this chapter:

(1) Given a distribution p which factors according to a graph $\mathcal{G}_{\preceq}$, what conditional independencies are implied by this factorization?

(2) How may we infer these independencies from the graph $\mathcal{G}_{\preceq}$?

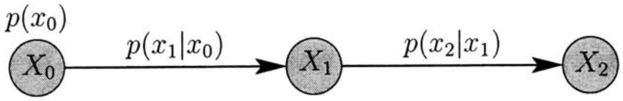


Figure 2.3: A simple Markov chain considered in Example 2.1.

(3) Conversely, given a distribution p satisfying specific conditional independencies, what does this imply about the structure of its factorization?

Example 2.1 (A Markov Chain).
Consider the multiscale model shown in Figure 2.3. Assuming that $X = \{X_0, X_1, X_2\}$ has a density p which admits a recursive factorization on $\mathcal{G}_{\preceq}$, then the following is true,

$$p(x_0, x_1, x_2) = p(x_0)p(x_1|x_0)p(x_2|x_1). \tag{2.15}$$

Given a density with such a special factorization, we first ask the question, "What conditional independencies do the random vectors $\{X_0, X_1, X_2\}$ exhibit?" Using (2.15) gives

$$p(x_0, x_2|x_1) = \frac{p(x_0, x_1, x_2)}{p(x_1)} = \frac{p(x_0)p(x_1|x_0)p(x_2|x_1)}{p(x_1)} = p(x_0|x_1)p(x_2|x_1), \tag{2.16}$$

thereby proving that X_0 and X_2 are conditionally independent given knowledge of X_1.

Now consider how we might infer this conditional independence property from the underlying graph $\mathcal{G}_{\preceq}$ shown in Figure 2.3. Notice that vertices 0 and 2 are separated when vertex 1 is removed from the graph. Therefore, in this example, the graphical notion of separation conveys the probabilistic notion of conditional independence. As Theorem 2.1 later shows, this is also true for any rooted tree $\mathcal{G}_{\preceq}$.

As the converse of the result discussed above, suppose that three random vectors $\{X_0, X_1, X_2\}$ have the property that X_0 and X_2 are conditionally independent given X_1. The conditional density $p(x_0, x_2|x_1)$ must therefore satisfy (2.16), and the joint density can be calculated as

$$\begin{aligned} p(x_0, x_1, x_2) &= p(x_0, x_2|x_1)p(x_1) = p(x_0|x_1)p(x_2|x_1)p(x_1) \\ &= p(x_0)p(x_1|x_0)p(x_2|x_1), \end{aligned}$$

showing that the random vectors $\{X_0, X_1, X_2\}$ admit a recursive factorization. Thus, in this example, by specifying a density p which satisfies the conditional independencies indicated by the graph $\mathcal{G}_{\preceq}$ (via its separation property), we obtain a density which recursively factors according to $\mathcal{G}_{\preceq}$. ◀

In order to characterize the conditional independencies exhibited by a multiscale model, it is necessary to consider specific subgraphs of the associated rooted tree $\mathcal{G}_{\preceq}$. To make this discussion more succinct, we introduce some additional notation. A calligraphic letter is used to represent a family of sets, *e.g.* $\mathcal{S} = \{S_1, \ldots, S_n\}$. The intersection and difference operations between a family of sets and a set is defined componentwise, *i.e.* $\mathcal{S} \cap S \triangleq \{S_1 \cap S, \ldots, S_n \cap S\}$ and $\mathcal{S} - S = \{S_1 - S, \ldots, S_n - S\}$. The union operation between two families is the usual set operation, *i.e.* if $\mathcal{R} = \{R_1, \ldots, R_m\}$, then $\mathcal{R} \cup \mathcal{S} = \{R_1, \ldots, R_m, S_1, \ldots, S_n\}$. Finally, the intersection and union operations on families are defined to be $\cap\mathcal{S} \triangleq \cap_{i=1}^n S_i$ and $\cup\mathcal{S} \triangleq \cup_{i=1}^n S_i$.

Now, suppose that a family of sets $\mathcal{S}$ satisfies the following property,

$$\begin{aligned} &\mathcal{S} - \cap\mathcal{S} = \{S_1^*, \ldots, S_n^*\}, \quad \text{and} \\ &S_i^* \neq \emptyset,\ \forall i \qquad S_i^* \cap S_j^* = \emptyset,\ \forall i \neq j. \end{aligned} \tag{2.17}$$

This property simply indicates that after removing the elements which are common to all sets in $\mathcal{S}$, the resulting sets are non-empty and have no common elements. Given a family which satisfies this property, the following statement concerning conditional independence is well-defined.

Definition 2.5 (Conditional Independence).
Given a family of sets $\mathcal{S} = \{S_1, \ldots, S_n\}$ with the property (2.17), we say that a set of vectors $\{X_v\}_{v \in \cup\mathcal{S}}$ satisfies conditional independence with respect to $\mathcal{S}$ if the following is true

$$p(x_{S_1^*}, \ldots, x_{S_n^*} | x_{\cap\mathcal{S}}) = \prod_{i=1}^{n} p(x_{S_i^*} | x_{\cap\mathcal{S}}). \tag{2.18}$$

The conditional independence of (2.18) is denoted by $\perp X_{\mathcal{S}}$. If $\cap\mathcal{S} = \emptyset$, then (2.18) is defined to be a statement of independence rather than conditional independence. ◀

For our purposes, the common intersection $\cap\mathcal{S}$ will in most cases be a singleton, but it is useful to express conditional independence in this type of set-theoretic language since the sets become increasingly complicated in subsequent sections.

For a specific illustration of Definition 2.5, consider the rooted tree shown in Figure 2.4. The three dashed contours divide the vertices into three sets which comprise the family $\mathcal{S}_1 = \{\{1,3,7,8\}, \{1,4,9,10\}, \{0,1,2,5,6,11,12,13,14\}\}$.[7] Notice that $\cap\mathcal{S}_1$ is the singleton $\{1\}$, and $\mathcal{S}_1 - \cap\mathcal{S}_1$ is the collection of subtrees obtained by removing vertex $\{1\}$ from the graph, specifically $\mathcal{S}_1 - \cap\mathcal{S}_1 = \{\{3,7,8\}, \{4,9,10\}, \{0,2,5,6,11,12,13,14\}\}$. Using the notation introduced in Section 2.2, we can write this collection of subtrees as $\mathcal{S}_1 - \cap\mathcal{S}_1 = \{S_3, S_4, S_1^c\}$, and we can also write the family of sets $\mathcal{S}_1$ as $\mathcal{S}_1 = \{\bar{S}_3, \bar{S}_4, S_1^c \cup \{1\}\}$.[8] The independence condition $\perp X_{\mathcal{S}_1}$ requires that the random vectors indexed by the sets S_3, S_4, and S_1^c be conditionally independent given X_1.

Consider now the following families of sets defined for all non-leaf vertices of a rooted tree $\mathcal{G}_{\preceq} = (V, E)$,

$$\mathcal{S}_{v_0} \triangleq \{\bar{S}_u\}_{u \in \chi(v_0)} \tag{2.19a}$$

$$\mathcal{S}_v \triangleq \{\bar{S}_u\}_{u \in \chi(v)} \cup \{S_v^c \cup \{v\}\}, \quad v \neq v_0. \tag{2.19b}$$

Each such family has the property that $\cap\mathcal{S}_v = \{v\}$, and using (2.19b), $\mathcal{S}_v - \cap\mathcal{S}_v = \{S_u\}_{u \in \chi(v)} \cup \{S_v^c\}$ are the subtrees obtained by removing vertex v from the graph $\mathcal{G}_{\preceq}$. The family $\mathcal{S}_1$ considered earlier and shown in Figure 2.4 provides one example of these types of sets.

Using the families (2.19), it is now straightforward to state the Markov property of interest. This property is a special case of the *directed global Markov property* defined in [71] for arbitrary directed acyclic graphs, but the following simpler version is sufficient for our purposes.

Definition 2.6 (The Global Markov Property).
Let $\mathcal{G}_{\preceq}$ be a rooted tree with vertices V. The collection of random vectors $\{X_v\}_{v \in V}$ are said to satisfy the *global Markov property at vertex* v if $\perp X_{\mathcal{S}_v}$ holds. Furthermore, $\{X_v\}_{v \in V}$ satisfies the *global Markov property* if $\perp X_{\mathcal{S}_v}$ holds for all non-leaf vertices v. ◀

[7] The significance of the notation $\mathcal{S}_1$ will be made clear later.

[8] Recall that for a rooted tree, the closure of the set S_v, $v \neq v_0$, is simply $\bar{S}_v = S_v \cup \{\pi(v)\}$.

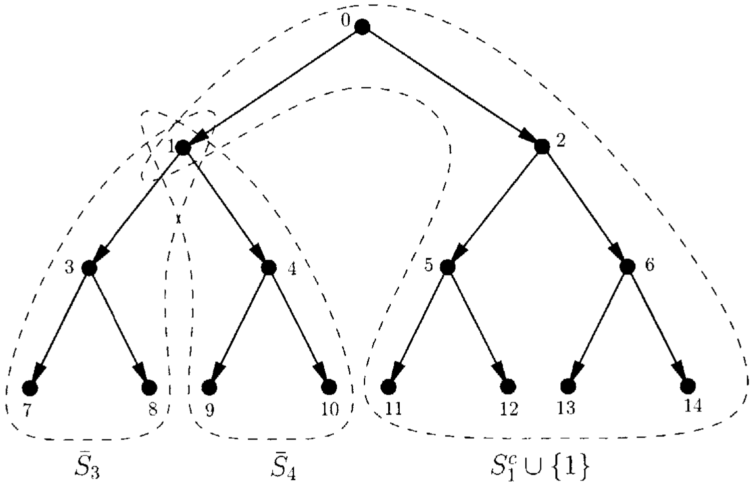


Figure 2.4. The dashed lines show the three sets of vertices required for the global Markov property to hold at vertex 1. Specifically, given the family of sets $\mathcal{S}_1 = \{\bar{S}_3, \bar{S}_4, S_1^c \cup \{1\}\}$ and the random vectors $X_0, \ldots, X_{14}$, the global Markov property holds at vertex 1 if $\perp X_{\mathcal{S}_1}$.

In summary, the global Markov property requires the following for every non-leaf vertex v: the random vectors indexed by the subtrees separated by v must be conditionally independent given X_v. In Figure 2.4, this property is satisfied at vertex 1 if $X_{\{3,7,8\}}$, $X_{\{4,9,10\}}$, and $X_{\{0,2,5,6,11,12,13,14\}}$ are jointly independent conditioned on the value of X_1. We call X_v a *state vector* if the global Markov property is satisfied at vertex v. This terminology is in keeping with the time-series literature where a state is a sufficient statistic of the past that makes the past and future conditionally independent.

As the following theorem evidences, the global Markov property completely characterizes the Markov properties of multiscale models. As proven in [71], the theorem applies more generally to any directed acyclic graphical model, but we only state the result pertaining to multiscale models.

Theorem 2.1 (Multiscale Models and the Global Markov Property).
$(X, \mathcal{G}_{\preceq})$ is a multiscale model if and only if $X = \{X_v\}$ satisfies the global Markov property.

Proof. See [71]. ■

Theorem 2.1 is important because it answers all three questions raised at the beginning of this section.

(1) Assuming that $(X, \mathcal{G}_{\preceq})$ is a multiscale model, the distribution $p(x)$ recursively factors according to $\mathcal{G}_{\preceq}$, and Theorem 2.1 states the specific conditional independencies that X satisfies.

(2) The conditional independencies $\perp X_{\mathcal{S}_v}$ are characterized by the structure of $\mathcal{G}_{\preceq}$. Specifically, the process indexed by the vertices separated by vertex v must be conditionally independent given X_v.

(3) The converse to Theorem 2.1 says that the independencies $\perp X_{\mathcal{S}_v}$ imply that $p(x)$ admits a recursive factorization according to $\mathcal{G}_{\preceq}$.

For the purposes of this thesis, we are most interested in the converse of this theorem. From the viewpoint of constructing a multiscale model, the converse indicates that if a set of vectors $\{X_v\}$

can be specified such that the global Markov property is satisfied with respect to $\mathcal{G}_{\preceq}$ then these vectors in fact correspond to a multiscale model on the graph $\mathcal{G}_{\preceq}$. Before discussing how to use this idea, though, it is necessary to analyze the global Markov property in more detail.

■ 2.5 The Reduced-Order Global Markov Property

The global Markov property is useful in the sense that it characterizes all possible multiscale models via a list of conditional independence requirements. For our needs, though, this list is a complicated set of coupled requirements. In this section, we show that by choosing an ordering on the non-leaf vertices of a rooted tree $\mathcal{G}_{\preceq}$, a smaller set of requirements are in fact equivalent to the global Markov property. We specifically address two important questions:

(1) What are the reduced-order sets which are equivalent to the sets required for the global Markov property?

(2) How may these sets be recognized from the graph $\mathcal{G}_{\preceq}$?

The following example provides some useful intuition concerning these reduced-order sets.

Example 2.2 (Reduced-Order Sets for Global Markovianity).
Consider the rooted tree shown in Figure 2.5(a). In this example, we examine the conditions required for the global Markov property to hold at vertices 0 and 1. According to Definition 2.6, we need $\perp X_{\mathcal{S}_0}$ and $\perp X_{\mathcal{S}_1}$ where

$$\mathcal{S}_0 = \{\bar{S}_1, \bar{S}_2\}$$
$$\mathcal{S}_1 = \{\bar{S}_3, \bar{S}_4, S_1^c \cup \{1\}\}$$

are shown by the dashed lines in Figures 2.5(a) and (b) respectively. As we demonstrate here, these independence conditions are coupled and include some redundancies. Specifically, there exists a family $\mathcal{R}_1 = \{\bar{S}_3, \bar{S}_4, R^*\}$ such that $R^* \subset S_1^c \cup \{1\}$, and the requirements $\perp X_{\mathcal{S}_0}$ and $\perp X_{\mathcal{R}_1}$ imply that $\perp X_{\mathcal{S}_0}$ and $\perp X_{\mathcal{S}_1}$ are also true.

To show this, define $R^* = \{0, 1\}$, so that $\perp X_{\mathcal{S}_0}$ and $\perp X_{\mathcal{R}_1}$ together require

$$p(x_{S_1}, x_{S_2}|x_0) = p(x_{S_1}|x_0)p(x_{S_2}|x_0) \tag{2.20a}$$
$$p(x_{S_3}, x_{S_4}, x_0|x_1) = p(x_{S_3}|x_1)p(x_{S_4}|x_1)p(x_0|x_1). \tag{2.20b}$$

Note that (2.20a) equivalently indicates that $p(x_{S_2}|x_0, x_{S_1}) = p(x_{S_2}|x_0)$. Using this fact along with (2.20b) and the chain rule for probabilities shows that the condition $\perp X_{\mathcal{S}_1}$ is satisfied,

$$\begin{aligned} p(x_{S_3}, x_{S_4}, x_{S_1^c}|x_1) &= p(x_{S_3}, x_{S_4}, \underbrace{x_0, x_{S_2}}_{x_{S_1^c}}|x_1) = p(x_{S_3}, x_{S_4}, x_0|x_1)p(x_{S_2}|\underbrace{x_{S_3}, x_{S_4}, x_1}_{x_{S_1}}, x_0) \\ &= p(x_{S_3}|x_1)p(x_{S_4}|x_1)p(x_0|x_1)p(x_{S_2}|x_0, x_1) \\ &= p(x_{S_3}|x_1)p(x_{S_4}|x_1)p(\underbrace{x_0, x_{S_2}}_{x_{S_1^c}}|x_1). \end{aligned}$$

It is important to note that this result depends on an explicit ordering of the vertices. Specifically, we have assumed that $\perp X_{\mathcal{S}_0}$ holds, and then, we specified a family $\mathcal{R}_1$ so that $\perp X_{\mathcal{S}_1}$ holds as well.

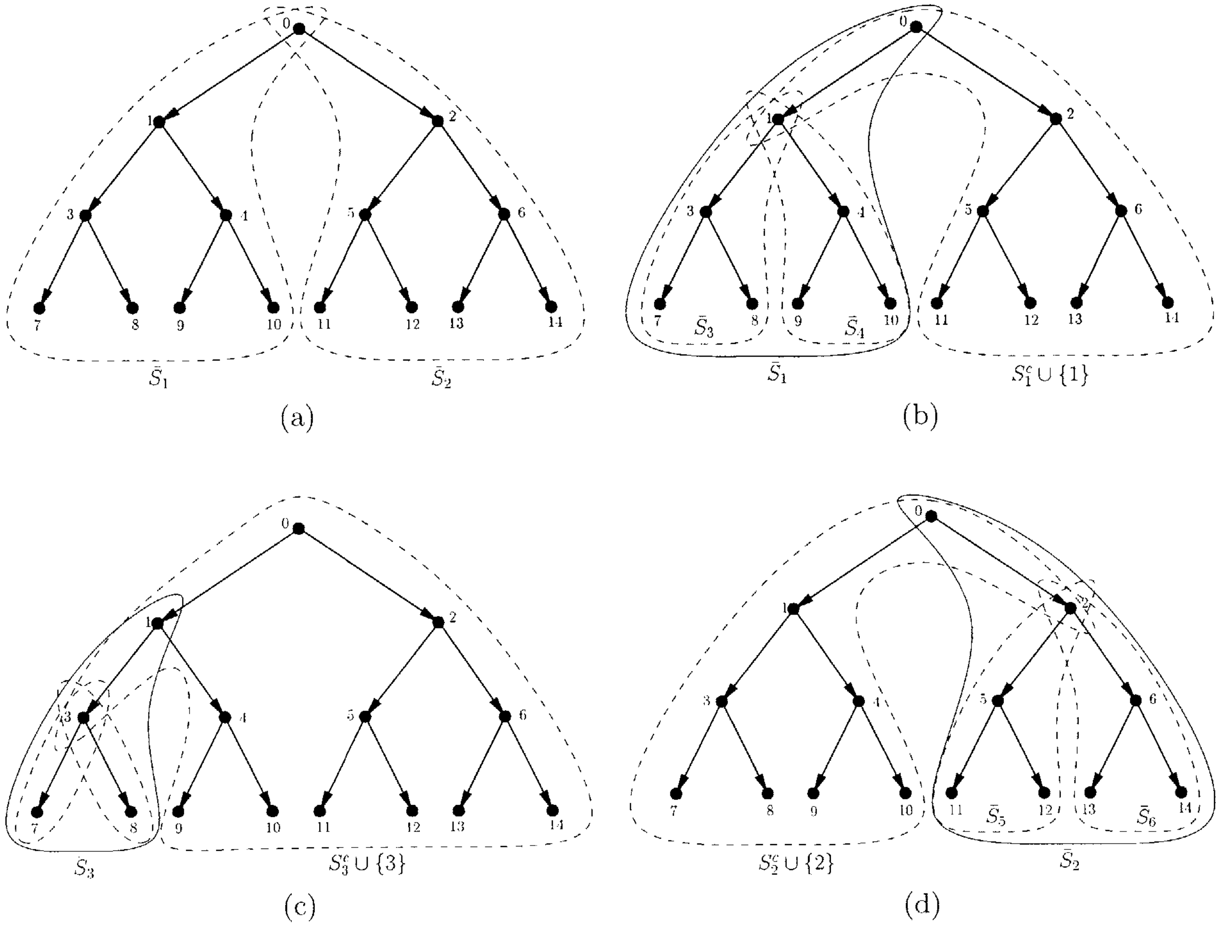


Figure 2.5. Graphical illustration of the sets required for the reduced-order global Markov property to be satisfied at vertices 0,1,2, and 3, for the ordering $(0, 1, 3, 2, \ldots)$. (a) The dashed lines show the family $\mathcal{S}_0$ necessary for the global Markov property to be satisfied at vertex 0. (b) The dashed lines show the family $\mathcal{S}_1$ necessary for the global Markov property to be satisfied at vertex 1. The solid line corresponds to the set $\bar{S}_1$ contained in $\mathcal{S}_0$. (c) The dashed lines show the family $\mathcal{S}_3$ necessary for the global Markov property to be satisfied at vertex 3. The solid line corresponds to the set $\bar{S}_3$ contained in $\mathcal{S}_1$. (d) The dashed lines show the family $\mathcal{S}_2$ necessary for the global Markov property to be satisfied at vertex 2. The solid line corresponds to the set $\bar{S}_2$ contained in $\mathcal{S}_0$.

More generally, the families $\mathcal{R}_v$ defined in this section require an ordering on the non-leaf vertices v of the graph.

Next, we ask the question of how to recognize this reduced-order property from the graph $\mathcal{G}_{\preceq}$. Recall that $\mathcal{R}_1 = \{\bar{S}_3, \bar{S}_4, \{0,1\}\}$ in this example. Examining Figure 2.5(b), note that all elements of $\mathcal{R}_1$ are contained within the set $\bar{S}_1 \in \mathcal{S}_0$, and in fact, $\mathcal{R}_1$ is the intersection of $\mathcal{S}_1$ with the set $\bar{S}_1$, thereby forming a refinement or partitioning of $\bar{S}_1$. The general characterization of the families $\mathcal{R}_v$ is slightly more complicated than this example implies, but the important intuition to take from this example is that the families $\mathcal{R}_v$ represent the portion of the sets in $\mathcal{S}_v$ contained within some boundary. There is no reason to include elements outside this boundary since other conditions $\perp X_{\mathcal{R}_{v_i}}$ already enforce these constraints. ◀

Let $(v_1, \ldots, v_m)$ be an arbitrary ordered set of all non-leaf vertices, and let $<$ denote the corresponding order, *i.e.* $v_1 < v_2 < \cdots < v_m$. For any distinct vertices $u \in V$ and $v \in (v_1, \ldots, v_m)$, there is a unique set $S \in \mathcal{S}_v$ which contains u. We henceforth use the notation S_v^u to represent this unique set, *i.e.*

$$S_v^u \triangleq S, \text{ where } S \in \mathcal{S}_v, \; u \in S, \; v \neq u. \tag{2.21}$$

For example, in Figure 2.4, the family $\mathcal{S}_1$ contains three sets, and we use the notation S_1^3 to indicate the unique set in $\mathcal{S}_1$ which contains vertex 3. Of course, there is also a unique set in $\mathcal{S}_1$ which contains vertex 7, *i.e.* S_1^7, and a unique set in $\mathcal{S}_1$ which contains vertex 8, *i.e.* S_1^8, and all of these sets are the same, *i.e.* $S_1^3 = S_1^7 = S_1^8$.

Now, let $v_i \in (v_2, \ldots, v_m)$ be a fixed vertex, and consider all families $\mathcal{S}_v$ with $v < v_i$. Each such family has a unique set $S_v^{v_i}$ which contains v_i. We define B_{v_i} to be the intersection of all such sets,

$$B_{v_i} \triangleq \bigcap_{v < v_i} S_v^{v_i}, \qquad v_i \in (v_2, \ldots, v_m). \tag{2.22}$$

This set B_{v_i} is what we call the "boundary of influence" for vertex v_i – an idea that we previously mentioned in Example 2.2. It is also convenient to define the following set

$$T_{v_i} \triangleq \{v | v \in B_{v_i}, v < v_i\}, \tag{2.23}$$

containing vertices in the boundary B_{v_i} that precede vertex v_i in the ordering.

Example 2.3 (Boundary Sets).
To provide some intuition for these boundary sets, consider again the rooted tree shown in Figure 2.5. This example assumes an ordering $(0, 1, 3, 2, \ldots)$ of the non-leaf vertices; the families $\mathcal{S}_0$,$\mathcal{S}_1$,$\mathcal{S}_3$, and $\mathcal{S}_2$ are respectively represented by the dashed lines in Figures 2.5(a),(b),(c), and (d).

The first boundary set B_1 is defined to be the set in $\mathcal{S}_0$ which contains vertex 1; this set is simply $\bar{S}_1$ and is represented by the solid line in Figure 2.5(b). The boundary set B_3 is the intersection of the set in $\mathcal{S}_0$ and the set in $\mathcal{S}_1$ which contain vertex 3, *i.e.* $B_3 = \bar{S}_1 \cap \bar{S}_3 = \bar{S}_3$. Similarly, the boundary B_2 is the intersection of $\bar{S}_2 \in \mathcal{S}_0$, $S_1^c \cup \{1\} \in \mathcal{S}_1$ and $S_3^c \cup \{3\} \in \mathcal{S}_3$. Examining Figure 2.5, it is easy to see that this intersection is $B_2 = \bar{S}_2$, as indicated by the solid line in Figure 2.5(d). In this case, the set $T_2 = \{0\}$ since vertex 0 is the only element of $B_2 = \bar{S}_2$ which precedes vertex 2 in the ordering.

The ordering $(0, 1, 3, 2, \ldots)$ considered in this example leads to a particularly simple set of boundaries, each of which correspond to subtrees of the graph. We will soon consider a more complicated example where the boundary sets are not so trivial. ◀

Given the definition (2.22) of the boundary sets, the reduced-order sets $\mathcal{R}_{v_i}$ are defined as follows,[9]

$$\mathcal{R}_{v_1} \triangleq \mathcal{S}_{v_1} \tag{2.24a}$$

$$\mathcal{R}_{v_i} \triangleq \mathcal{S}_{v_i} \cap B_{v_i}, \quad i = 2, \ldots, m. \tag{2.24b}$$

In Figures 2.5(a)–(d), the reduced-order sets are respectively $\mathcal{R}_0 = \mathcal{S}_0$, $\mathcal{R}_1 = \{\bar{S}_3, \bar{S}_4, \{0,1\}\}$, $\mathcal{R}_3 = \{\bar{S}_7, \bar{S}_8, \{1,3\}\}$, and $\mathcal{R}_2 = \{\bar{S}_5, \bar{S}_6, \{0,2\}\}$.

Definition 2.7 (The Reduced-Order Global Markov Property).
Random vectors $\{X_v\}$ are said to satisfy the *reduced-order global Markov property* with respect to the ordering $(v_1, \ldots, v_m)$ of non-leaf vertices if $\perp X_{\mathcal{R}_{v_i}}$ for $i = 1, \ldots, m$. ◀

Before stating the main result of this section, it is useful to further examine the boundary sets B_{v_i} and the families $\mathcal{R}_{v_i}$. We do this first through an example and then more formally through the two results provided in Proposition 2.2.

Example 2.4 (Characterization of Boundary and Reduced-Order Sets).
In this example, we graphically illustrate two important properties of the boundary and reduced-order sets, utilizing Figures 2.6 and 2.7, and we assume the ordering $(3, 6, 1, 0, \ldots)$ on the non-leaf vertices. Figure 2.6(a) shows the reduced-order family $\mathcal{R}_3$, and since vertex 3 is first in the ordering, we have by definition $\mathcal{R}_3 = \mathcal{S}_3$. For the sake of discussion in this example, we denote the sets in any family $\mathcal{R}_v$ by $R_v^{(1)}$, $R_v^{(2)}$, ... Figure 2.6(b) illustrates both the family $\mathcal{S}_6$ (dashed) and its associated boundary B_6 (solid), and Figure 2.6(c) shows the resulting intersection $\mathcal{R}_6 = \mathcal{S}_6 \cap B_6$. In a similar manner, Figure 2.6(d) shows the family $\mathcal{S}_1$ and boundary B_1, and Figure 2.7(a) shows the resulting intersection $\mathcal{R}_1 = \mathcal{S}_1 \cap B_1$. Figures 2.7(b),(c) show similar results for vertex 0.

This example shows that, depending on the chosen ordering, the boundary sets can be much more complicated than the sets previously provided in Example 2.3. More importantly, it emphasizes an important property of the boundary sets. Notice in Figure 2.6(b) that the boundary set B_0 is equal to one of the reduced-order sets $R_3^{(3)}$. In addition, $B_1 = R_6^{(3)}$ and $B_0 = R_1^{(3)}$ as indicated respectively in Figures 2.6(d) and 2.7(b). Consequently, each boundary set is actually a reduced-order set, which then implies that any family $\mathcal{R}_v = \mathcal{S}_v \cap B_v = \mathcal{S}_v \cap R$ is refinement of another reduced-order set R. The progression of sets shown in Figures 2.6(a),(c) and 2.7(a),(c) illustrate this property of refinement. This result is more formally stated in Proposition 2.2.

Another important property of the boundary sets can be seen by examining Figures 2.6(b),(d), and 2.7(b). Notice that the vertices not contained within the boundary set can be partitioned into disjoint sets which are graphically separated by the boundary set. In Figure 2.6(b), there is only one such set $\{7, 8\}$, and in Figure 2.6(d), there are two such sets $\{7, 8\}$ and $\{13, 14\}$. Figure 2.7(d) illustrates this idea for the boundary B_0. Notice that the vertex $t_1 = 1$ in Figure 2.7(d) separates the subgraph induced by $A_1 \cup \{t_1\}$ from the rest of the graph; similarly, $t_2 = 6$ separates the subgraph induced by $A_2 \cup \{t_2\}$ from the rest of the graph. Furthermore, we have $T_0 = \{t_1, t_2\}$ in this case. This result is more formally stated in Proposition 2.2. ◀

[9] Notice that this notation is somewhat misleading in that the family $\mathcal{R}_{v_i}$ is not fixed and explicitly depends on the ordering $(v_1, \ldots, v_m)$. In contrast, the family $\mathcal{S}_{v_i}$ is fixed. For simplicity, we do not mention the associated ordering when discussing the families $\mathcal{R}_{v_i}$ since it will be assumed.

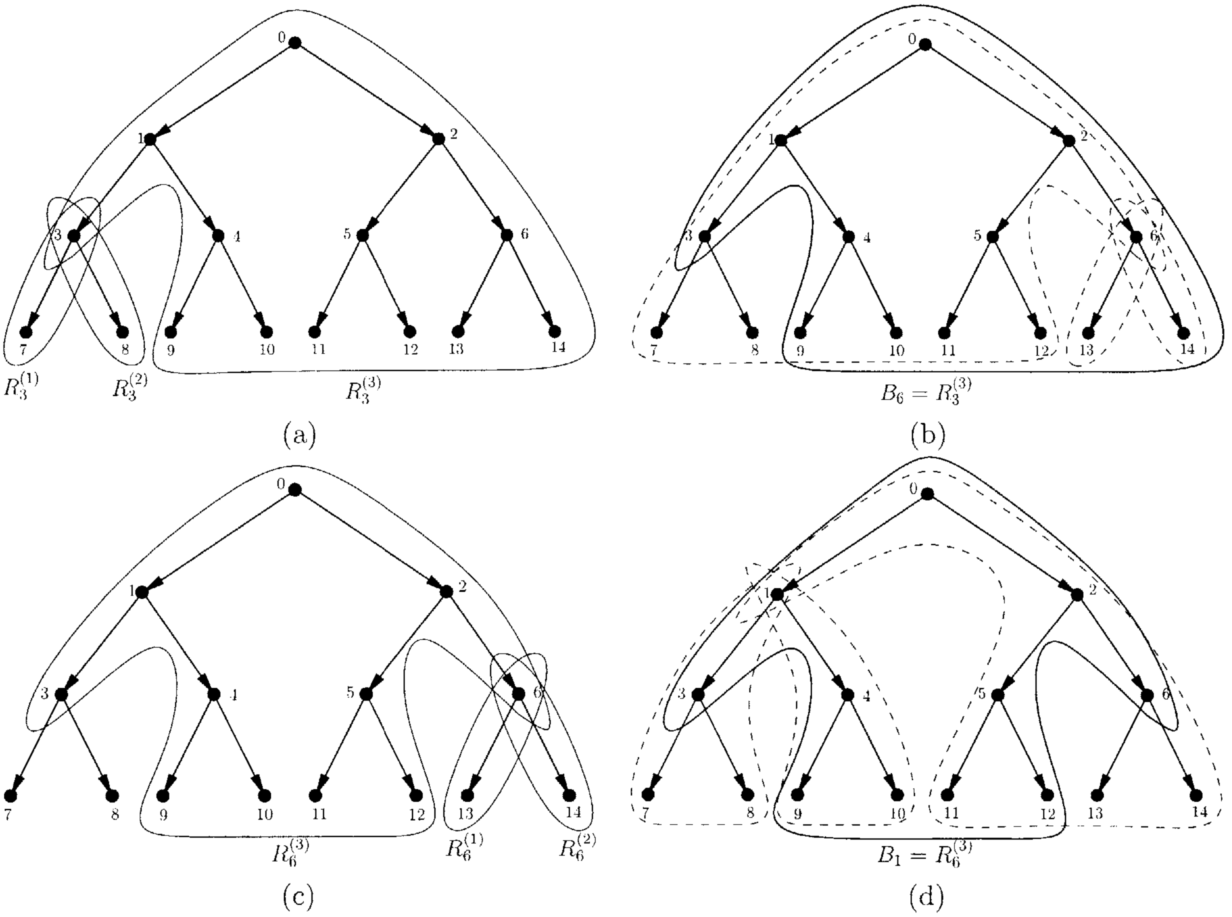


Figure 2.6. Graphical illustration of the result provided in Proposition 2.2. (a),(c) Show the sets contained in the two families $\mathcal{R}_3$ and $\mathcal{R}_6$ respectively. (b),(d) Shows how the reduced-order families $\mathcal{R}_6$ and $\mathcal{R}_1$ respectively are obtained by partitioning a "previous" reduced-order set.

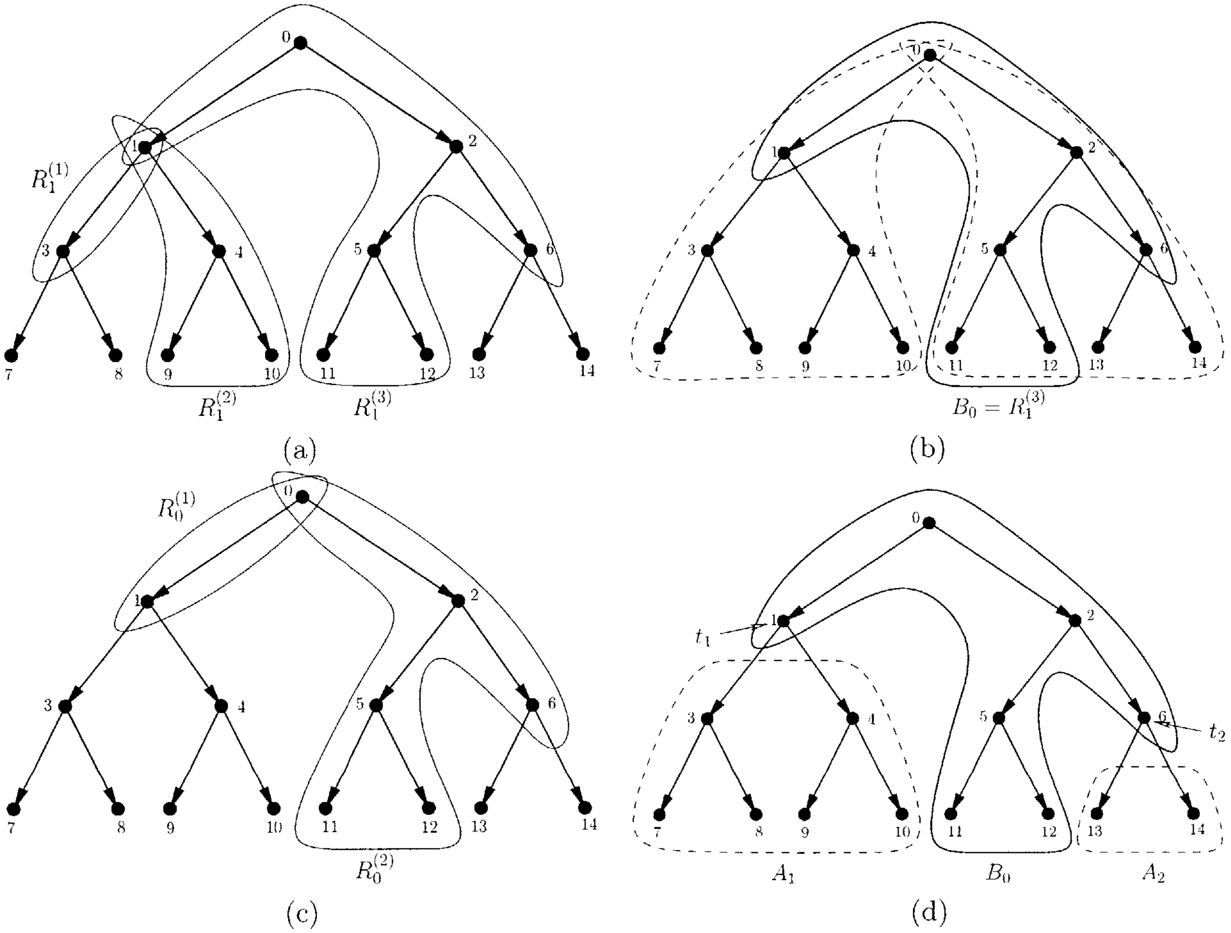


Figure 2.7. Graphical illustration of the result provided in Proposition 2.2. (a),(c) Show the sets contained in the two families $\mathcal{R}_1$ and $\mathcal{R}_0$ respectively. (b) Shows how the reduced-order family $\mathcal{R}_0$ is obtained by partitioning a "previous" reduced-order set. (d) Illustrates the result given in the second part of Proposition 2.2; specifically, the vertices satisfy $V = A_1 \cup A_2 \cup B_0$, and for $i = 1, 2$, the vertex t_i separates the subgraph induced by $A_i \cup \{t_i\}$ from the rest of the graph.

Proposition 2.2 (Characterization of Boundary and Reduced-Order Sets).
Let $(v_1, \ldots, v_m)$ be an ordering of the non-leaf vertices of a graph $\mathcal{G}_{\preceq} = (V, E)$.

(1) For any $i = 2, \ldots, m$, the set B_{v_i} defined in (2.22) is equal to some set $R \in \mathcal{R}_v$ with $v < v_i$. Consequently, $\mathcal{R}_{v_i} = \mathcal{S}_{v_i} \cap B_{v_i} = \mathcal{S}_{v_i} \cap R$ is a partitioning of the set R.

(2) For any $v_i \in (v_2, \ldots, v_m)$, suppose $T_{v_i} = \{t_1, \ldots, t_n\}$. The vertices V may be written as the union of $n+1$ disjoint sets $A_1, \ldots, A_n$, and B_{v_i}, where the subgraph induced by $A_j \cup \{t_j\}$ is separated from the rest of the graph by vertex t_j.

Proof. See Appendix A.2. ■

Using the second part of Proposition 2.2, it is straightforward to prove the following theorem which indicates that the reduced-order global Markov property is equivalent to the global Markov property.

Theorem 2.2 (Equivalence of Global and Reduced-Order Global Markov Properties).
Random vectors $\{X_v\}$ satisfy the global Markov property if and only if they satisfy the reduced-order global Markov property.

Proof. See Appendix A.2. ■

The great benefit of this result is that the number of conditional independencies required to satisfy the global Markov property has been reduced. However, this result has not solved another problem – the conditional independencies are still coupled. For example, Figures 2.7(a) and (c) show that the independence conditions $\perp X_{\mathcal{R}_1}$ and $\perp X_{\mathcal{R}_0}$ place constraints on the vectors X_0 and X_1. From a design standpoint, this means that X_0 and X_1 must be specified simultaneously in order to jointly satisfy these constraints. The results provided in the following section show that the degree of coupling may be reduced if a different but related problem is considered.

■ 2.6 Marginalization-Invariant Markovianity

In this section, we continue our discussion of the Markov properties associated with multiscale models, but we consider a slightly different problem. Suppose that $\{X_v\}_{v \in V}$ is now a collection of random vectors associated with a graph $\mathcal{G}_{\preceq}$ and having an <u>arbitrary</u> density p. Consider another density q defined as follows,

$$q(x) \triangleq \prod_{v \in V} p(x_v | x_{\pi(v)}). \tag{2.25}$$

By definition, q recursively factors according to $\mathcal{G}_{\preceq}$, and therefore $(X, \mathcal{G}_{\preceq})$ is a multiscale model under the density q. Of course, if p also recursively factors according to $\mathcal{G}_{\preceq}$, then $q = p$, but for an arbitrary density p, this will not necessarily be true.

Suppose, however, we select a subset of vertices $M \subset V$ and consider the marginal densities $p(x_M)$ and $q(x_M)$. The goal of this section is to characterize the densities p for which $p(x_M) = q(x_M)$ is satisfied. Specifically, what conditional independence properties must p satisfy (which in turn says something about the factorization of p) such that $p(x_M) = q(x_M)$. If $M \subsetneq V$, we show that the number of conditional independencies which p must satisfy in order for $p(x_M) = q(x_M)$ is generally smaller than the number of conditional independencies exhibited by q. The following example illustrates this idea.

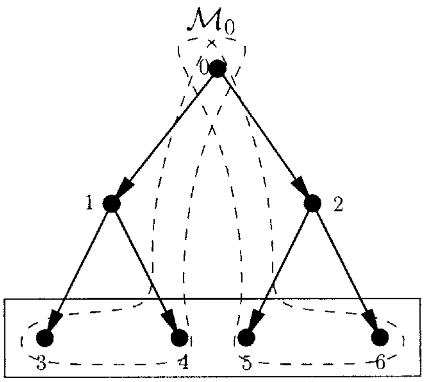


Figure 2.8. The rooted tree $\mathcal{G}_{\preceq}$ considered in Example 2.5. The dashed lines show the two sets included in the family $\mathcal{M}_0$, and the solid box represents the set $\{3,4,5,6\}$ for which the marginal constraint $p(x_3,x_4,x_5,x_6) = q(x_3,x_4,x_5,x_6)$ must be satisfied.

Example 2.5 (Sets for Marginalization-Invariant Markovianity).
Consider the graph shown in Figure 2.8. Using the results provided in the previous section, the reduced-order sets for this example, assuming the ordering $(0,1,2)$, are as follows,

$$\begin{aligned}\mathcal{R}_0 &= \{\bar{S}_1, \bar{S}_2\} = \{\{0,1,3,4\},\{0,2,5,6\}\}\\ \mathcal{R}_1 &= \{\{1,3\},\{1,4\},\{1,0\}\}\\ \mathcal{R}_2 &= \{\{2,5\},\{2,6\},\{2,0\}\}.\end{aligned}$$

Suppose we are given a density $p(x_0,\ldots,x_6)$, and further, suppose that we define a density $q(\cdot)$ which recursively factors according to the graph $\mathcal{G}_{\preceq}$ shown in Figure 2.8, *i.e.*

$$q(x_0,\ldots,x_6) \triangleq p(x_0)p(x_1|x_0)p(x_2|x_0)p(x_3|x_1)p(x_4|x_1)p(x_5|x_2)p(x_6|x_2). \tag{2.26}$$

Theorem 2.2 implies that the conditions $\perp X_{\mathcal{R}_0}$, $\perp X_{\mathcal{R}_1}$, and $\perp X_{\mathcal{R}_2}$ on the density $p(\cdot)$ are sufficient to ensure that $q(\cdot) = p(\cdot)$.

For the purpose of this example, we are interested in examining the conditions on $p(\cdot)$ for which the densities agree on the finest scale, *i.e.* under which $q(x_3,x_4,x_5,x_6) = p(x_3,x_4,x_5,x_6)$ is satisfied. We propose here a family $\mathcal{M}_0$ such that the independence conditions $\perp X_{\mathcal{M}_0}$ are less stringent than the conditions $\perp X_{\mathcal{R}_0}$, and we show that together the conditions $\perp X_{\mathcal{M}_0}$, $\perp X_{\mathcal{R}_1}$, and $\perp X_{\mathcal{R}_2}$ on $p(\cdot)$ ensure that $q(\cdot)$ has the correct marginal density. Specifically, define the family $\mathcal{M}_0$ (graphically displayed in Figure 2.8) as follows

$$\mathcal{M}_0 \triangleq \{\{0,3,4\},\{0,5,6\}\}. \tag{2.27}$$

To begin, first suppose $\perp X_{\mathcal{R}_2}$ is satisfied under the density $p(\cdot)$, *i.e.*

$$\begin{aligned}p(x_0,x_2,x_5,x_6) &= p(x_0,x_5,x_6|x_2)p(x_2) = p(x_0|x_2)p(x_5|x_2)p(x_6|x_2)p(x_2)\\ &= p(x_0)p(x_2|x_0)p(x_5|x_2)p(x_6|x_2).\end{aligned}$$

Substituting this expression into (2.26) gives

$$\begin{aligned}q(x_0,\ldots,x_6) &= p(x_1|x_0)p(x_3|x_1)p(x_4|x_1)p(x_0,x_2,x_5,x_6)\\ &= p(x_1|x_0)p(x_3|x_1)p(x_4|x_1)p(x_0)p(x_5,x_6|x_0)p(x_2|x_0,x_5,x_6).\end{aligned} \tag{2.28}$$

Next, suppose $\perp X_{\mathcal{R}_1}$ is satisfied under the density $p(\cdot)$, *i.e.*

$$\begin{aligned} p(x_0, x_1, x_3, x_4) &= p(x_0, x_3, x_4|x_1)p(x_1) = p(x_0|x_1)p(x_3|x_1)p(x_4|x_1)p(x_1) \\ &= p(x_0)p(x_1|x_0)p(x_3|x_1)p(x_4|x_1). \end{aligned}$$

Substituting this expression into (2.28) gives

$$\begin{aligned} q(x_0, \ldots, x_6) &= p(x_0, x_1, x_3, x_4)p(x_5, x_6|x_0)p(x_2|x_0, x_5, x_6) \\ &= p(x_0)p(x_3, x_4|x_0)p(x_5, x_6|x_0)p(x_2|x_0, x_5, x_6)p(x_1|x_0, x_3, x_4). \end{aligned} \tag{2.29}$$

Finally, the condition $\perp X_{\mathcal{M}_0}$ requires

$$p(x_0, x_3, x_4, x_5, x_6) = p(x_0)p(x_3, x_4|x_0)p(x_5, x_6|x_0),$$

and using this in (2.29) gives

$$\begin{aligned} q(x_0, \ldots, x_6) &= p(x_0, x_3, x_4, x_5, x_6)p(x_2|x_0, x_5, x_6)p(x_1|x_0, x_3, x_4) \\ &= p(x_3, x_4, x_5, x_6)p(x_2|x_0, x_5, x_6)p(x_1|x_0, x_3, x_4)p(x_0|x_3, x_4, x_5, x_6). \end{aligned} \tag{2.30}$$

Examining (2.30), note that integrating out the variables x_2, x_1, and x_0 (in that order) proves the claim.

Therefore, if the density $q(\cdot)$ need only satisfy the marginal requirement $q(x_3, x_4, x_5, x_6) = p(x_3, x_4, x_5, x_6)$ then $\perp X_{\mathcal{R}_0}$, $\perp X_{\mathcal{R}_1}$, and $\perp X_{\mathcal{R}_2}$ impose more constraints than necessary for the task; in this case, the requirements $\perp X_{\mathcal{M}_0}$, $\perp X_{\mathcal{R}_1}$, and $\perp X_{\mathcal{R}_2}$ are sufficient. Intuitively, this result implies that the set of vertices $\{3, 4, 5, 6\} \subsetneq V$ must be included in all of the relevant independence constraints, while vertices not included in this set may be safely removed from some of the constraints. ◀

2.6.1 Definition of Marginalization-Invariant Markovianity and Main Theorem

We now define the sets which characterize the marginalization-invariant Markov property. Let M, which we call the *marginalization constraint set*, be a specified subset of the vertices V, and let $(v_1, \ldots, v_m)$ be an ordering of the non-leaf vertices of $\mathcal{G}_{\preceq}$. Consider the sequence of sets,

$$M^{(0)} \triangleq M \tag{2.31a}$$

$$M^{(i)} \triangleq M^{(i-1)} \cup \{v_i\}, \quad i = 1, \ldots, m. \tag{2.31b}$$

The families of interest here are defined as follows,[10]

$$\mathcal{M}_{v_i} \triangleq \mathcal{R}_{v_i} \cap M^{(i)}, \quad i = 1, \ldots, m. \tag{2.32}$$

Notice that this definition resembles the definition of the families $\mathcal{R}_{v_i}$ in (2.24), where in this case, $M^{(i)}$ takes on the role of a "boundary" set. In particular, $M^{(1)} = M \cup \{v_1\}$ can be considered a boundary of importance for vertex v_1, and $\mathcal{M}_{v_1} = \mathcal{R}_{v_1} \cap M^{(1)}$ is a partitioning of that boundary. The analogy ends there, however, because $M^{(i)} \subset M^{(i+1)}$ is an increasing sequence of sets, whereas

[10] As with the reduced-order families $\mathcal{R}_{v_i}$, the families $\mathcal{M}_{v_i}$ explicitly depend on the ordering $(v_1, \ldots, v_m)$.

the boundary sets satisfy $B_{v_i} \supset B_{v_j}$ for some $v_j > v_i$ (see the first part of Proposition 2.2). Consequently, the sets $M^{(i)}$ function somewhat differently than the boundary sets B_{v_i}.

Some intuition about (2.32) can be gained by examining the first and last families in the ordering, *i.e.* $\mathcal{M}_{v_1}$ and $\mathcal{M}_{v_m}$. Note that if $M^{(1)} = M \cup \{v_1\}$ is a strict subset of the vertices V, then $\mathcal{M}_{v_1} = \mathcal{R}_{v_1} \cap M^{(1)} = \mathcal{S}_{v_1} \cap M^{(1)}$ is a "smaller" family than $\mathcal{R}_{v_1}$. That is, $\perp X_{\mathcal{M}_{v_1}}$ contains fewer conditional independence requirements than $\perp X_{\mathcal{R}_{v_1}}$. At the other extreme, if we assume that M contains all of the leaf vertices of $\mathcal{G}_{\preceq}$ plus possibly some subset of non-leaf vertices, then we have $M^{(m)} = V$. As a result, $\mathcal{M}_{v_m} = \mathcal{R}_{v_m} \cap V = \mathcal{R}_{v_m}$, implying that $\perp X_{\mathcal{M}_{v_m}}$ and $\perp X_{\mathcal{R}_{v_m}}$ impose equivalent constraints. Intuitively, the remaining families $\mathcal{M}_{v_i}$ with $v_1 < v_i < v_m$ contain some tradeoff between the marginal requirement specified by the set M and the Markov requirements specified by the families $\mathcal{R}_{v_i}$.

As an example of this fact, consider the graph shown in Figure 2.8 with the ordering $(0, 1, 2)$ and with $M = \{3, 4, 5, 6\}$. The sequence of sets given by (2.31) are $M^{(0)} = \{3, 4, 5, 6\}$, $M^{(1)} = \{0, 3, 4, 5, 6\}$, $M^{(2)} = \{0, 1, 3, 4, 5, 6\}$, and $M^{(3)} = \{0, 1, 2, 3, 4, 5, 6\}$. The families $\mathcal{M}_{v_i}$ are then given by

$$\begin{aligned}
\mathcal{M}_0 &= \mathcal{R}_0 \cap M^{(1)} = \{\{0,3,4\},\{0,5,6\}\} \\
\mathcal{M}_1 &= \mathcal{R}_1 \cap M^{(2)} = \mathcal{R}_1 \\
\mathcal{M}_2 &= \mathcal{R}_2 \cap M^{(3)} = \mathcal{R}_2.
\end{aligned}$$

These are precisely the sets considered previously in Example 2.5.

Having defined the families $\mathcal{M}_{v_i}$ in (2.32), we now consider a new Markov property, namely the *marginalization-invariant Markov property.* The significance of this name will be made clear shortly in Theorem 2.3.

Definition 2.8 (The Marginalization-Invariant Markov Property).
Random vectors $\{X_v\}_{v \in V}$ are said to satisfy the *marginalization-invariant Markov property* with respect to the ordering $(v_1, \ldots, v_m)$ of non-leaf vertices and with respect to the set $M \subset V$ if $\perp X_{\mathcal{M}_{v_i}}$ for $i = 1, \ldots, m$. ◄

Example 2.6 (Sets for Marginalization-Invariant Markovianity).
As an additional illustration of the sets required for marginalization-invariant Markovianity, we consider the graph shown in Figure 2.9, where in this case $M = \{7, \ldots, 14\}$ contains all leaf vertices. We choose the ordering $(0, 1, 3, 2, \ldots)$[11] on the non-leaf vertices. Recall that this graph and ordering was previously considered in Example 2.3 and Figure 2.5. The reduced-order family $\mathcal{R}_0$ is represented by the dashed lines in Figure 2.5(a), while the families $\mathcal{R}_1$, $\mathcal{R}_3$, and $\mathcal{R}_2$ are the intersection of the dashed contours with the solid contour in each of the Figures 2.5(b)–(d) respectively.

Given the ordering $(0, 1, 3, 2, \ldots)$, the first few sets $M^{(i)}$ in (2.31) are given by: $M^{(1)} = \{7, \ldots, 14, 0\}$, $M^{(2)} = \{7, \ldots, 14, 0, 1\}$, $M^{(3)} = \{7, \ldots, 14, 0, 1, 3\}$, and $M^{(4)} = \{7, \ldots, 14, 0, 1, 3, 2\}$. The families $\mathcal{M}_0$, $\mathcal{M}_1$, $\mathcal{M}_3$, and $\mathcal{M}_2$ obtained from the intersection of a reduced-order set with the appropriate set $M^{(j)}$ are graphically depicted in Figure 2.9. ◄

The following theorem indicates that any tree-indexed process $(X, \mathcal{G}_{\preceq})$ which satisfies the marginalization-invariant Markov property also possesses a very specific marginal invariance. This

[11] The notation $(0, 1, 3, 2, \ldots)$ indicates that we are only interested in the first four vertices of the ordering, *i.e.* 0, 1, 3, and 2. The remaining vertices in the ordering can be arbitrary.

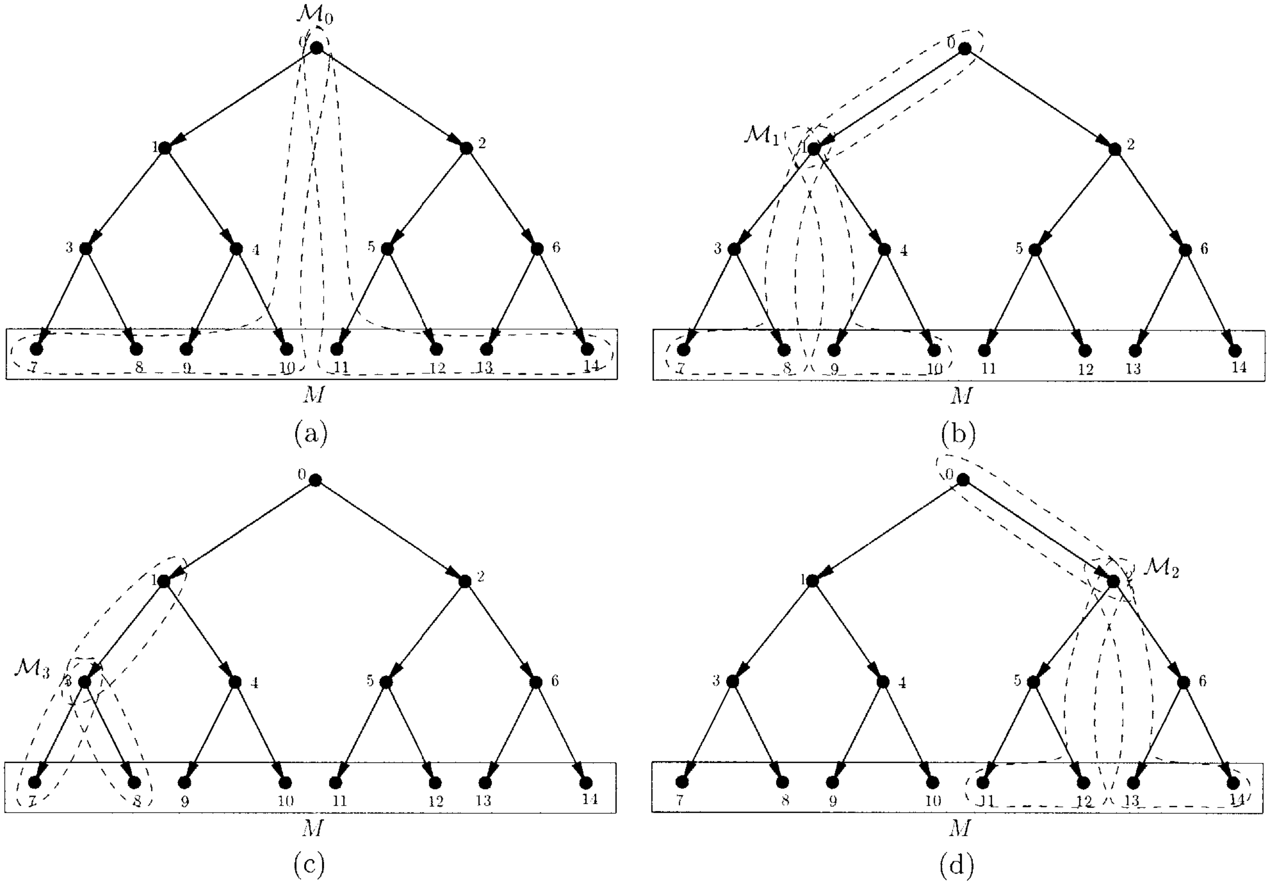


Figure 2.9. Graphical illustration of the sets required for the marginalization-invariant Markov property to hold, assuming an ordering $(0, 1, 3, 2, \ldots)$ on the non-leaf vertices and assuming that $M = \{7, \ldots, 14\}$. The dashed contours define the sets which comprise the families: (a) $\mathcal{M}_0$ (b) $\mathcal{M}_1$ (c) $\mathcal{M}_3$ (d) $\mathcal{M}_2$.

theorem is the most important result provided in this chapter, and it is the basis of our remaining discussion on multiscale realization.

Theorem 2.3 (Significance of the Marginalization-Invariant Markov Property).
Suppose the random vectors $\{X_v\}_{v\in V}$ admit a probability density $p(\cdot)$, and define the density $q(x) \triangleq \prod_{v\in V} p(x_v|x_{\pi(v)})$. If $\{X_v\}$ satisfies the marginalization-invariant Markov property with respect to a specified set M and any ordering $(v_1,\dots,v_m)$ of the non-leaf vertices of $\mathcal{G}_{\preceq}$, then $q(x_M) = p(x_M)$.

Proof. See Section 3.9.1. ■

This is the same result proven in Example 2.5 for a special case.

Notice that Theorem 2.2 now follows directly from Theorem 2.3. Letting $M = V$, the families $\mathcal{M}_{v_i}$ are equal to the families $\mathcal{R}_{v_i}$, implying that the marginalization-invariant and reduced-order Markov conditions are equivalent in this case. Theorem 2.3 indicates that if these marginalization-invariant (reduced-order) conditions are satisfied, then $p(x_V) = q(x_V)$ recursively factors according to $\mathcal{G}_{\preceq}$, and using Theorem 2.1, the global Markov property must be satisfied.

■ 2.6.2 Two Special Cases

In the previous section, marginalization-invariant Markovianity was defined for arbitrary orderings $(v_1,\dots,v_m)$ of the non-leaf vertices. As one might expect, the families $\mathcal{M}_{v_i}$ and the associated conditional independence constraints can be rather complicated for arbitrary orderings. In this section, we limit the types of orderings $(v_1,\dots,v_m)$ to so-called *top-down* and *bottom-up* orderings, with the goal of providing a simpler characterization of the families $\mathcal{M}_{v_i}$. Most of the examples considered henceforth will use one of the following two types of orderings.

Definition 2.9 (Top-down Ordering).
An ordering $(v_1,\dots,v_m)$ of the non-leaf vertices of a graph $\mathcal{G}_{\preceq}$ is called *top-down* if there exists no $v_j, v_k \in (v_1,\dots,v_m)$ with $v_k > v_j$ and $v_k \prec v_j$.[12] ◄

Definition 2.10 (Bottom-up Ordering).
An ordering $(v_1,\dots,v_m)$ of the non-leaf vertices of a graph $\mathcal{G}_{\preceq}$ is called *bottom-up* if there exists no $v_j, v_k \in (v_1,\dots,v_m)$ with $v_k < v_j$ and $v_k \prec v_j$. ◄

If we imagine constructing a top-down ordering by adding non-leaf vertices one at a time in a sequential fashion, then Definition 2.9 indicates that a vertex v can be added only if its parent has already appeared in the ordering; consequently, the root vertex must appear first in the ordering. Similarly, constructing a bottom-up ordering in a sequential fashion requires that a vertex v can be added only if all of its children have already appeared in the ordering; in this case, the root vertex must appear last in the ordering. We choose the names top-down and bottom-up because they are visually suggestive when the graph $\mathcal{G}_{\preceq}$ is drawn with the root vertex on top and the leaf vertices on bottom, as illustrated in most of the graphs shown so far. As an example, $(0,1,3,2,5,4,6)$ is a top-down ordering for the graph shown in Figure 2.9, while $(3,6,5,2,4,1,0)$ is a bottom-up ordering.

We now characterize the families $\mathcal{M}_{v_i}$ for these two types of orderings. In order to simplify this characterization, we additionally assume that the marginalization constraint set M is precisely

[12] Recall that $<$ is the ordering associated with $(v_1,\dots,v_m)$, and $\prec$ is the partial ordering defined in Section 2.2.

the set of all leaf vertices of the specified graph $\mathcal{G}_{\preceq}$.[13] As we later discuss, this is a reasonable assumption for many of the realization problems of interest.

Proposition 2.3 (Marginalization-Invariant Markovianity and a Top-Down Ordering). *Suppose the marginalization constraint set M is equal to all leaf vertices of a graph $\mathcal{G}_{\preceq}$, and let $(v_1, \ldots, v_m)$ be a top-down ordering of the non-leaf vertices. Then, the families $\mathcal{M}_{v_i}$ may be written as follows:*

$$\mathcal{M}_{v_0} - \{v_0\} = \{L_v\}_{v \in \chi(v_0)}, \tag{2.33a}$$

$$\mathcal{M}_{v_i} - \{v_i\} = \{L_v\}_{v \in \chi(v_i)} \cup \{\pi(v_i)\}, \quad v_i \neq v_0. \tag{2.33b}$$

Proof. See Appendix A.3. ■

For simplicity of the expression on the right-hand side of (2.33), we have chosen to characterize $\mathcal{M}_{v_i} - \{v_i\}$ instead of $\mathcal{M}_{v_i}$.[14] Suppose that vertex v_i has q children $\chi(v_i)$, then the right side of (2.33b) is the collection of $q+1$ sets: the q sets L_v, each containing the leaf vertices that descend from a child v of v_i, and the $(q+1)^{st}$ set containing only the parent of v_i. As an example, we previously considered a top-down ordering in Figure 2.9, where we illustrated the first four families $\mathcal{M}_{v_i}$ for the ordering $(0, 1, 3, 2, \ldots)$.

The characterization of the families $\mathcal{M}_{v_i}$ is slightly more complicated for a bottom-up ordering than for a top-down ordering. It is useful for our purposes to introduce the following operator which maps subsets of vertices to subsets of vertices,

$$\min_{\mathcal{G}_{\preceq}}(A) \triangleq \{v \in A | \text{there does not exist a } u \in A \text{ with } u \prec v\}. \tag{2.34}$$

Therefore, $\min_{\mathcal{G}_{\preceq}}(A)$ intuitively consists of the elements of A "closest" to the root vertex. More precisely, $v \in \min_{\mathcal{G}_{\preceq}}(A)$ if and only if $m(v) < m(u)$ for all vertices $u \in A$ which are comparable to v with respect to $\preceq$.

Proposition 2.4 (Marginalization-Invariant Markovianity and a Bottom-Up Ordering). *Suppose the marginalization constraint set M is equal to all leaf vertices of a graph $\mathcal{G}_{\preceq}$, and let $(v_1, \ldots, v_m)$ be a bottom-up ordering of the non-leaf vertices. Then, the families $\mathcal{M}_{v_i}$ may be written as follows:*

$$\mathcal{M}_{v_0} = \{\{v_0, v\}\}_{v \in \chi(v_0)}, \tag{2.35a}$$

$$\mathcal{M}_{v_i} = \{\{v_i, v\}\}_{v \in \chi(v_i)} \cup \left\{\min_{\mathcal{G}_{\preceq}}\left(M^{(i)}\right)\right\}, \quad v_i \neq v_0. \tag{2.35b}$$

Proof. See Appendix A.3. ■

Assuming that vertex v_i has q children, the right side of (2.35b) is the collection of $q+1$ sets: q of these are doubleton sets each containing v_i and a single child of v_i. The remaining set, containing the smallest elements of $M^{(i)}$, is more difficult to visualize. First of all, it must contain v_i. To see this, recall that $M^{(i)}$ contains the set M (leaf vertices in this case) plus any vertices $v_j \preceq v_i$.

[13] If M contains any non-leaf vertices, the families $\mathcal{M}_{v_i}$ explicitly depend on which non-leaf vertices are contained in M, regardless of the ordering. By restricting M to the set of leaf vertices, we can completely characterize the families $\mathcal{M}_{v_i}$ for top-down and bottom-up orderings.

[14] Recall that the operation $\mathcal{M}_{v_i} - \{v_i\}$ removes the common element $\{v_i\}$ from each set in the family $\mathcal{M}_{v_i}$.

By the nature of the bottom-up ordering, $v_i \in \min_{\mathcal{G}_{\preceq}}(M^{(i)})$ because it is smaller (with respect to $\prec$) than any of its descendants, each of which must be included in $M^{(i)}$, and because there is no element smaller (with respect to $\prec$) than v_i in $M^{(i)}$. The remaining elements of $\min_{\mathcal{G}_{\preceq}}(M^{(i)})$ contain the vertices in $M^{(i)}$ "closest" to the root vertex and such that no element is a descendent of another. As an example, Figures 2.10 illustrate the families $\mathcal{M}_{v_i}$ for the bottom-up ordering $(3, 6, 5, 2, 4, 1, 0)$. The family $\mathcal{M}_0$, which is not shown, is given by $\mathcal{M}_0 = \{\{0,1\},\{0,2\}\}$.

■ 2.7 The Multiscale Realization Problem

As mentioned previously, the ultimate goal of studying the conditional independence properties of multiscale models is to provide additional machinery for the multiscale realization problem. Theorem 2.3 provides significant guidance because it offers a prescription or recipe for solving the realization problem. At this point, the connection between Theorem 2.3 and the realization problem might not be at all obvious to the reader – we relate the two in the following section.

■ 2.7.1 Multiscale Realization and Theorem 2.3

Suppose a random process Y is observed and can be described by a probability density p, and suppose we map this process to a subset M of the vertices of a rooted tree $\mathcal{G}_{\preceq}$ such that $Y = \{X_v\}_{v \in M}$.[15] Consequently, we have defined a density $p(x_M)$. See Figure 2.11(a) for an example of a typical mapping.

The goal of the multiscale realization problem is to: (1) specify a density $q(x_V)$, which recursively factors according to a graph $\mathcal{G}_{\preceq}$ and (2) satisfy the marginal requirement $q(x_M) = p(x_M)$. The connection between the realization problem and Theorem 2.3 is that the theorem allows one to characterize all possible models $q(\cdot)$ which satisfy the above two requirements. In order to apply the theorem, though, we must have a density $p(x_V)$ over all vertices V, rather than simply the subset M.[16] This suggests that we must somehow specify an *extension* of the given density $p(x_M)$ to a *complete* density $p(x_V)$, where $p(x_V)$ agrees with the marginal $p(x_M)$, *i.e.* $\int_{x_{V-M}} p(x_V)dx_V = p(x_M)$. Then, if any complete density $p(x_V)$ satisfies the marginalization-invariant Markov property with respect to the set M, Theorem 2.3 indicates that the marginal constraint $p(x_M) = q(x_M)$ will be satisfied, and consequently, $q(x_V) \triangleq \prod_{v \in V} p(x_v|x_{\pi(v)})$ is a solution to the realization problem.

More specifically, the realization problem can be viewed in two steps:

(1) Specify any conditional density $\bar{p}(x_{V-M}|x_M)$, such that the random vectors $\{X_v\}_{v \in V}$ (under the complete density $p(x_V) = \bar{p}(x_{V-M}|x_M)p(x_M)$) satisfy the marginalization-invariant Markov property. Notice that the conditional density $\bar{p}(x_{V-M}|x_M)$ provides the required extension from $p(x_M)$ to $p(x_V)$, and it implicitly represents the degrees of freedom which we have in the realization problem.

(2) Based on this complete density $p(x_V)$, define $q(x_V) \triangleq \prod_{v \in V} p(x_v|x_{\pi(v)})$.

[15] The choice of the particular rooted tree $\mathcal{G}_{\preceq}$, the chosen subset M, and the mapping of the process to the vertices M are all important issues. For now, we assume that we have made adequate choices for a given problem at hand.

[16] As we later discuss in Section 2.7.2, Theorem 2.3 may be applied to the realization problem even if the entire density $p(x_V)$ is not known. We will in fact show that it is sufficient to have a specific set of marginals of $p(x_V)$, but for the sake of discussion here, we assume that it is necessary to know the entire density $p(x_V)$.

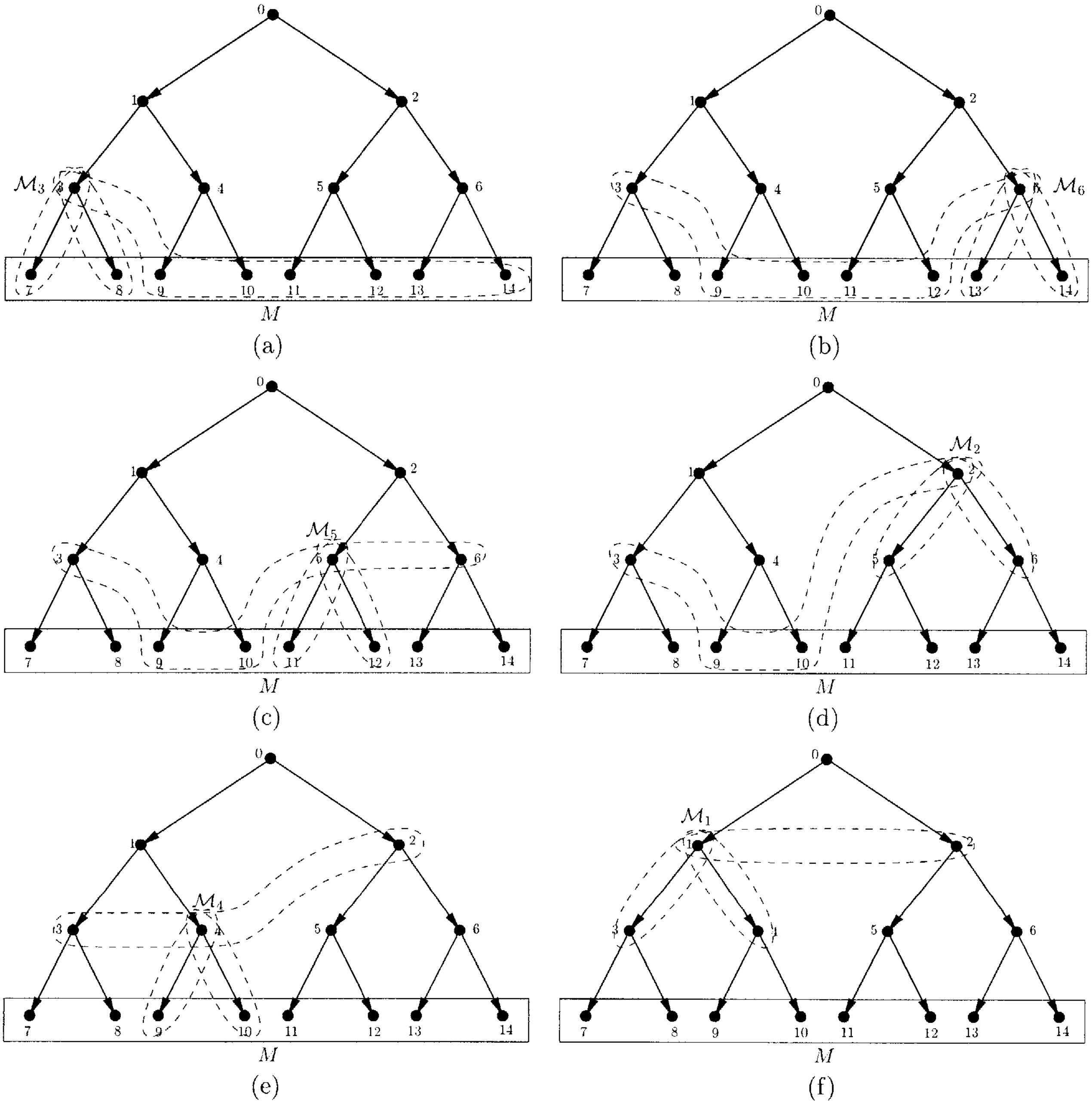
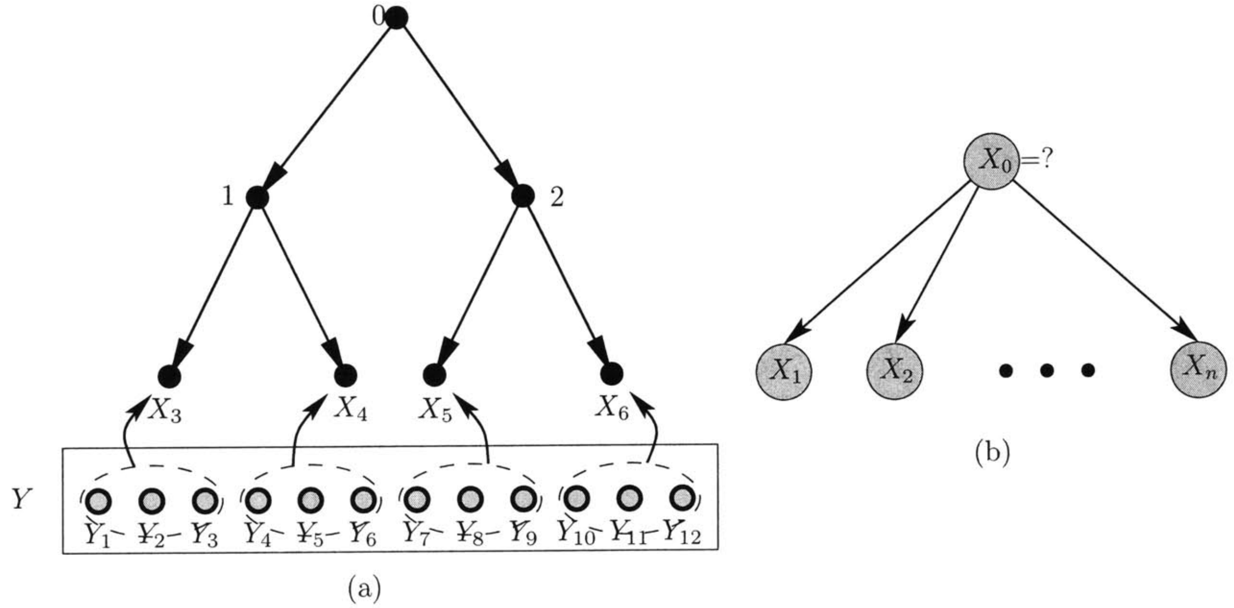


Figure 2.10. Graphical illustration of the sets required for the marginalization-invariant Markov property to hold, assuming a bottom-up ordering $(3, 6, 5, 2, 4, 1, 0)$ on the non-leaf vertices and assuming that $M = \{7, \ldots, 14\}$. The dashed contours define the sets which comprise the families: (a) $\mathcal{M}_3$ (b) $\mathcal{M}_6$ (c) $\mathcal{M}_5$ (d) $\mathcal{M}_2$ (e) $\mathcal{M}_4$ (f) $\mathcal{M}_1$.


0
1
2
X_3
X_4
X_5
X_6
Y
Y_1 Y_2 Y_3 Y_4 Y_5 Y_6 Y_7 Y_8 Y_9 Y_{10} Y_{11} Y_{12}
X_0=?
X_1
X_2
X_n

(a)

(b)

Figure 2.11. (a) An example mapping of an observed process Y to a subset of the vertices of a rooted tree. (b) One of the simplest multiscale realization problems. Specifically, given a density $p(x_1, \ldots, x_n)$, how should X_0 be defined such that $X_1, X_2, \ldots, X_n$ are conditionally independent given knowledge of X_0? A trivial solution to this problem is given by $X_0 = (X_1, \ldots, X_{n-1})^T$.

Hence, we view the realization problem as a search for a "valid" conditional density $\bar{p}(x_{V-M}|x_M)$, *i.e.* a conditional density which allows $p(x_M) = q(x_M)$ to be satisfied. We use the notation $\bar{p}(x_{V-M}|x_M)$ to remind the reader that, unlike $p(x_M)$, this conditional density is unspecified and must be designed as part of the realization problem.

As one might suspect, there are an infinite number of "valid" conditional densities, and most of these are trivial solutions to the problem. Suppose, for example, we have the problem graphically displayed in Figure 2.11(b) where we want to specify a random vector X_0 such that n random vectors $X_1, \ldots, X_n$ are made conditionally independent given knowledge of X_0. Choosing X_0 to be composed of the vectors $X_1, \ldots, X_{n-1}$ constitutes a valid solution to this problem, and similarly, any $(n-1)$-sized subset of the vectors $X_1, \ldots, X_n$ is a valid solution. For a more complicated graph $\mathcal{G}_{\preceq}$ than the one shown in Figure 2.11(b), if we allow the dimensionality of the random vectors X_v, $v \in V - M$, to be arbitrary, then there exist many trivial solutions of this nature.

Recall that the fundamental reason for using multiscale models (at least for the Gaussian case) is the efficiency of estimation tasks, and this efficiency is achieved only when the dimensionality of each vector X_v is much smaller than the dimensionality of the observed random process. In order to avoid trivial solutions in which some or all vectors X_v have large dimensionality, we must impose constraints on these dimensions. In doing so, however, there is a distinct possibility that no exact solutions to the realization problem will exist. While choosing these constraints is an important and challenging problem, it is not the focus of this thesis; instead, one of two approaches is taken:

(1) If an obvious non-trivial solution to the realization problem exists, then we choose the dimensions of X_v, $v \in V - M$, to match this solution. This will be the case for the examples

considered later in Section 2.8, where we are more interested in academic examples that provide intuition.

(2) Based on either an arbitrary choice or some knowledge about the problem, the dimensions of X_v are fixed. This is the view we will take in subsequent chapters when discussing approximate multiscale models.

In the remainder of this chapter, our approach to finding a valid and non-trivial solution to the multiscale realization problem is (given Theorem 2.3) a very natural one. In particular, an exact multiscale model can be realized in a sequential fashion by specifying the random vectors X_v, $v \in V - M$, one at a time. The fact that a multiscale model can be realized in a successive fashion is due to two inherent properties of the families $\mathcal{M}_{v_i}$ and the independence constraints $\perp X_{\mathcal{M}_{v_i}}$. We discuss these important properties in the following section.

2.7.2 Sequential Realization of Multiscale Models

An important issue associated with marginalization-invariant Markovianity, Theorem 2.3, and consequently with the realization problem is that of coupled conditional independence constraints. This is an issue with which we have been concerned throughout this chapter. In this section and the next, we show that for all problems of interest to us, the constraints associated with the marginalization-invariant Markov property can be *ordered* so that a sequential realization procedure is possible.

Example 2.7 (Sequential Realization).
As an illustration of what we mean by sequential realization, consider again Example 2.6 and Figure 2.9, where we imposed a marginal constraint on the leaves of the multiscale model, *i.e.* $p(x_7, \ldots, x_{14}) = q(x_7, \ldots, x_{14})$, and where we chose the ordering $(0, 1, 3, 2, \ldots)$ on the non-leaf vertices. As discussed in the previous section, we can use Theorem 2.3 to realize a multiscale model $q(\cdot)$ which satisfies $p(x_7, \ldots, x_{14}) = q(x_7, \ldots, x_{14})$ if we can find a conditional density $\bar{p}(x_0, \ldots, x_6 | x_7, \ldots, x_{14})$ such that $\{X_0, \ldots, X_{14}\}$ (under the complete density $p(x_0, \ldots, x_{14})$) satisfies the marginalization-invariant Markov property. This example shows that the marginalization-invariant Markov property can be satisfied in a sequential manner, which immediately implies that an exact multiscale model can be realized in a sequential manner.

As demonstrated by the family $\mathcal{M}_0$ in Figure 2.9(a), the independence constraint $\perp X_{\mathcal{M}_0}$ is satisfied if $X_{\{7,\ldots,10\}}$ and $X_{\{11,\ldots,14\}}$ are conditionally independent given $X_0 = x_0$, under some density $p(x_0, x_7, \ldots, x_{14})$. Therefore, by specifying $\bar{p}(x_0 | x_7, \ldots, x_{14})$ such that $X_{\{7,\ldots,10\}}$ and $X_{\{11,\ldots,14\}}$ are conditionally independent given $X_0 = x_0$ (under the joint density $p(x_0, x_7, \ldots, x_{14})$), we will satisfy $\perp X_{\mathcal{M}_0}$. Notice that in this first step we use the given marginal density $p(x_7, \ldots, x_{14})$ to design the new vector X_0.

Continuing in a similar fashion, random vector X_1 must satisfy the condition $\perp X_{\mathcal{M}_1}$, where the family $\mathcal{M}_1$ is shown in Figure 2.9(b). Here, the role of X_1 is to make $X_{\{7,8\}}$, $X_{\{9,10\}}$, and X_0 conditionally independent under some density $p(x_0, x_1, x_7, x_8, x_9, x_{10})$. Notice that in designing X_0 in the previous step, we have specified the density $p(x_0, x_7, \ldots, x_{14})$, and by marginalizing out the variables $x_{11}, x_{12}, x_{13}, x_{14}$, we get the density $p(x_0, x_7, x_8, x_9, x_{10})$. Then, in order to design random vector X_1, we must specify a conditional density $\bar{p}(x_1 | x_0, x_7, x_8, x_9, x_{10})$ so that we satisfy $\perp X_{\mathcal{M}_1}$.

This second step in the realization procedure raises a very important point. Namely, in order to satisfy the constraint $\perp X_{\mathcal{M}_1}$, we only need to specify the density $p(x_0, x_1, x_7, x_8, x_9, x_{10})$, not the larger density $p(x_0, x_1, x_7, \ldots, x_{14})$. More generally, this step in the realization procedure illustrates the fact that Theorem 2.3 does not impose conditions on the full density $p(x)$; instead, it imposes conditions on a set of marginals of $p(x)$. In terms of this example, this means that we never really specify the complete conditional density $\bar{p}(x_0, \ldots, x_6|x_7, \ldots, x_{14})$ (or the complete joint density $p(x_0, \ldots, x_{14})$), but instead, we specify the relevant pieces of this conditional density, *i.e* $\bar{p}(x_0|x_7, \ldots, x_{14})$, $\bar{p}(x_1|x_0, x_7, x_8, x_9, x_{10})$, *etc.*, so that the necessary marginal densities satisfy the marginalization-invariant Markov property.

The realization procedure continues in a similar fashion by successively defining the vectors X_3, X_2, and so forth. Figure 2.12 provides a graphical illustration of the steps involved in realizing the multiscale model in this example, where we have chosen the ordering $(0, 1, 3, 2, 4, 5, 6)$ on the non-leaf vertices. Notice that at each step of the procedure we use a previously specified density to design a new vector X_v; in so doing, we ensure that all of the marginal densities are *consistent*, *i.e.* all densities have marginals that agree on the variables which they have in common. Also notice that the structure of the block diagram in Figure 2.12 remains the same for any top-down ordering. That is, for any top-down ordering, the same set of conditional densities are designed. This property, however, does not hold for bottom-up orderings, *i.e.* different bottom-up orderings require different conditional densities to be specified explicitly. ◀

Having provided an example of a sequential realization procedure, we now define and discuss two properties of the families $\mathcal{M}_{v_i}$ which permit such a procedure. We then propose a general algorithm for realizing multiscale models in a sequential fashion, and we subsequently show that the algorithm is well-defined and generates an exact multiscale model when the marginalization constraint set M contains only the leaf vertices of a rooted tree $\mathcal{G}_{\preceq}$.

The preceding example, along with Figure 2.12, provides some intuition for our discussion. In particular, notice that the families $\mathcal{M}_{v_i}$ possess two properties which are essential to the existence of a sequential realization procedure. The first property, which we call *orderability* and formally define in Definition 2.11, is a set-theoretic property of the families $\mathcal{M}_{v_i}$ that allows the vectors X_v, $v \notin M$, to be designed in a successive fashion. This property is evident in Example 2.7 because the vectors X_0, X_1, X_3, X_2, *etc.* are designed successively with respect to previously defined vectors. The second property, which we call a *nesting* property, ensures that each of the marginal densities can be specified in a consistent fashion. This property is evident in Example 2.7 because we are able to design new densities from previously specified densities, thereby avoiding conflicting marginal densities, *i.e.* $p(x_7, \ldots, x_{14})$ is used to design $p(x_0, x_7, \ldots, x_{14})$; $p(x_0, x_7, \ldots, x_{14})$ is used to design $p(x_0, x_1, x_7, \ldots, x_{10})$; $p(x_0, x_1, x_7, \ldots, x_{10})$ is used to design $p(x_1, x_3, x_7, x_8)$; *etc.* We now discuss these two properties in more detail.

The following definition characterizes the previously mentioned orderability property. For the purposes of this definition, suppose that each of the families $\mathcal{M}_1, \mathcal{M}_2, \ldots, \mathcal{M}_m$ satisfy the property given in (2.17) and satisfy $\cap\mathcal{M}_i \neq \emptyset$ for $1 \leq i \leq m$. This simply ensures that the conditional independence constraints $\perp X_{\mathcal{M}_1}, \perp X_{\mathcal{M}_2}, \ldots, \perp X_{\mathcal{M}_m}$ are valid. We choose to state this set-theoretic property in terms of a generic set of families $\mathcal{M}_i$, $1 \leq i \leq m$, and a set $M \subseteq V$[17]. The reader should associate the family $\mathcal{M}_i$ with the marginalization-invariant family $\mathcal{M}_{v_i}$ and associate the set M with the marginalization constraint set M. As we later show in Proposition 2.5, the families

[17] The set M must be included in the definition of the orderability property. From the standpoint of realization, M is the marginalization constraint set associated with the specified density $p(x_M)$. To ensure that the families $\mathcal{M}_i$ are orderable and hence a sequential realization procedure is possible, we must require that the constraints $\perp X_{\mathcal{M}_i}$ do not involve a design variable X_v with $v \in M$. Consequently, our definition of orderability includes this set M.

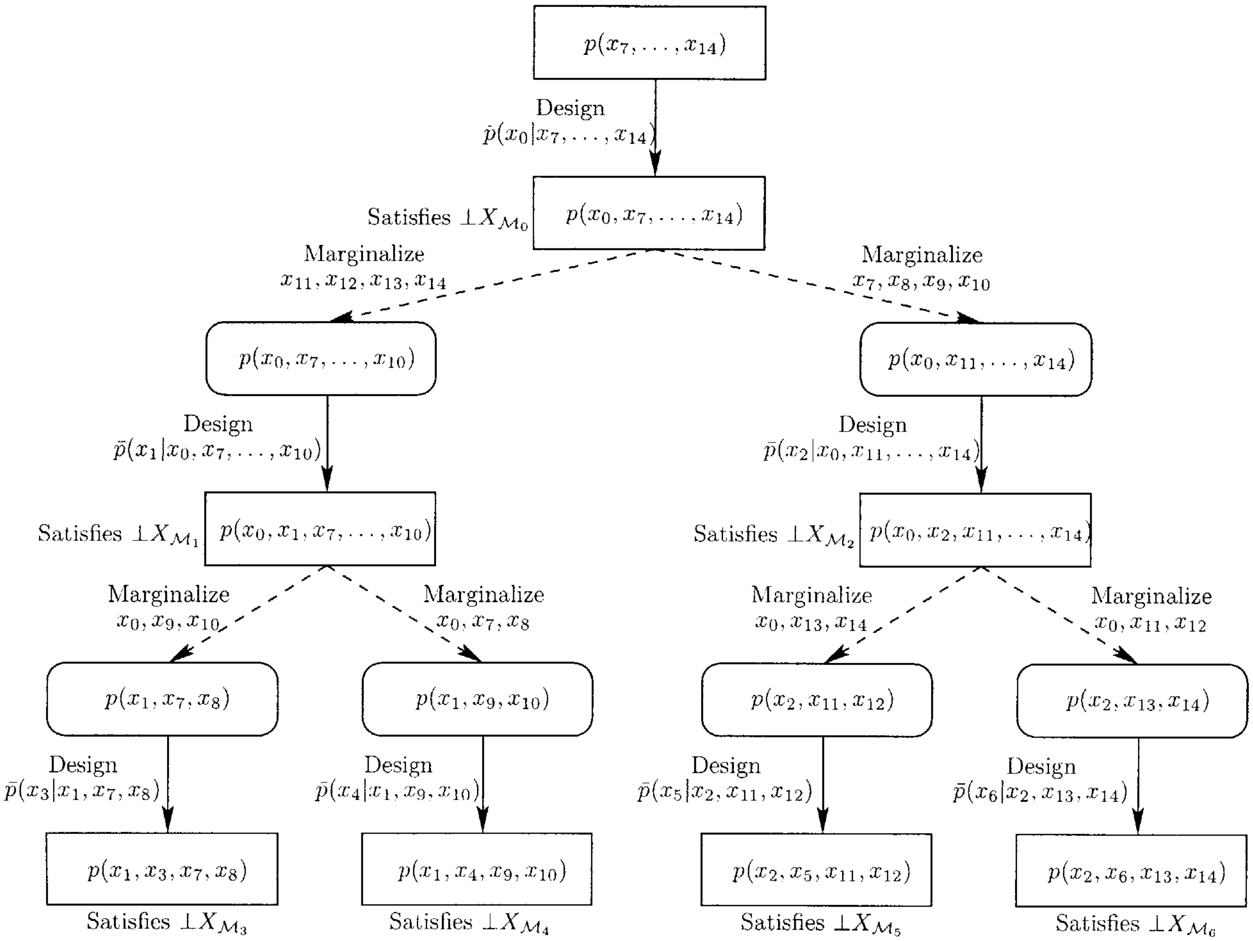

$p(x_7, \ldots, x_{14})$
Design
$\bar{p}(x_0|x_7, \ldots, x_{14})$
Satisfies $\perp X_{\mathcal{M}_0}$
$p(x_0, x_7, \ldots, x_{14})$
Marginalize
$x_{11}, x_{12}, x_{13}, x_{14}$
Marginalize
x_7, x_8, x_9, x_{10}
$p(x_0, x_7, \ldots, x_{10})$
$p(x_0, x_{11}, \ldots, x_{14})$
Design
$\bar{p}(x_1|x_0, x_7, \ldots, x_{10})$
Design
$\bar{p}(x_2|x_0, x_{11}, \ldots, x_{14})$
Satisfies $\perp X_{\mathcal{M}_1}$
$p(x_0, x_1, x_7, \ldots, x_{10})$
Satisfies $\perp X_{\mathcal{M}_2}$
$p(x_0, x_2, x_{11}, \ldots, x_{14})$
Marginalize
x_0, x_9, x_{10}
Marginalize
x_0, x_7, x_8
Marginalize
x_0, x_{13}, x_{14}
Marginalize
x_0, x_{11}, x_{12}
$p(x_1, x_7, x_8)$
$p(x_1, x_9, x_{10})$
$p(x_2, x_{11}, x_{12})$
$p(x_2, x_{13}, x_{14})$
Design
$\bar{p}(x_3|x_1, x_7, x_8)$
Design
$\bar{p}(x_4|x_1, x_9, x_{10})$
Design
$\bar{p}(x_5|x_2, x_{11}, x_{12})$
Design
$\bar{p}(x_6|x_2, x_{13}, x_{14})$
$p(x_1, x_3, x_7, x_8)$
$p(x_1, x_4, x_9, x_{10})$
$p(x_2, x_5, x_{11}, x_{12})$
$p(x_2, x_6, x_{13}, x_{14})$
Satisfies $\perp X_{\mathcal{M}_3}$
Satisfies $\perp X_{\mathcal{M}_4}$
Satisfies $\perp X_{\mathcal{M}_5}$
Satisfies $\perp X_{\mathcal{M}_6}$

Figure 2.12. Block diagram illustrating the steps involved in a sequential realization of the multiscale model considered in Example 2.7, with the ordering $(0, 1, 3, 2, 4, 5, 6)$ on the non-leaf vertices. The rectangular boxes contain the densities needed to satisfy the conditions $\perp X_{\mathcal{M}_{v_i}}$, while the rounded boxes contain intermediate densities which result from a marginalization step. The dashed arrows indicate a marginalization of a density, while the solid arrows represent a design step where a conditional density must be specified.

$\mathcal{M}_{v_i}$ satisfy the following property when the marginalization constraint set M contains only leaf vertices.

Definition 2.11 (Ordered Conditional Independence Constraints).
The conditional independence constraints $\perp X_{\mathcal{M}_1}, \perp X_{\mathcal{M}_2}, \ldots, \perp X_{\mathcal{M}_m}$ are said to be *ordered* with respect to a set $M \subseteq V$ if for $j = 1, \ldots, m$, the following conditions are satisfied:

$$\begin{aligned} &\cap \mathcal{M}_j \not\subset M \\ &\cap \mathcal{M}_j \not\subset \cup \mathcal{M}_k \quad \text{for all } k < j. \end{aligned} \qquad \blacktriangleleft$$

To apply Definition 2.11 to a particular example, consider again Example 2.7 where the first four families in the ordering $(0, 1, 3, 2, \ldots)$ are $\mathcal{M}_0 = \{\{0, 7, 8, 9, 10\}, \{0, 11, 12, 13, 14\}$, $\mathcal{M}_1 = \{\{0, 1\}, \{1, 7, 8\}, \{1, 9, 10\}\}$, $\mathcal{M}_3 = \{\{3, 7\}, \{3, 8\}, \{1, 3\}\}$, and $\mathcal{M}_2 = \{\{0, 2\}, \{2, 11, 12\}, \{2, 13, 14\}\}$. This sequence of families satisfies Definition 2.11, with respect to the marginalization constraint set $M = \{7, \ldots, 14\}$, since $\cap \mathcal{M}_1 = \{1\}$ is not a subset of $\cup \mathcal{M}_0$ or M; $\cap \mathcal{M}_3 = \{3\}$ is not a subset of $\cup \mathcal{M}_0$, $\cup \mathcal{M}_1$, or M; and $\cap \mathcal{M}_2 = \{2\}$ is not a subset of $\cup \mathcal{M}_0$, $\cup \mathcal{M}_1$, $\cup \mathcal{M}_3$, or M.

From the perspective of sequential realization, Definition 2.11 ensures that each constraint $\perp X_{\mathcal{M}_j}$ introduces a new *design* variable. For example, in Example 2.7, $X_{\cap \mathcal{M}_0} = X_0$, $X_{\cap \mathcal{M}_1} = X_1$, $X_{\cap \mathcal{M}_3} = X_3$, and $X_{\cap \mathcal{M}_2} = X_2$ are the design variables in the realization problem since their role is to make certain sets of variables conditionally independent. The fact that the families $\perp X_{\mathcal{M}_0}$, $\perp X_{\mathcal{M}_1}$, $\perp X_{\mathcal{M}_3}$, and $\perp X_{\mathcal{M}_2}$ are ordered implies that the design variables X_0, X_1, X_3, and X_2 may be defined successively. This follows from the fact that X_1 is not associated with the conditions $\perp X_{\mathcal{M}_0}$ or the given density $p(x_M)$; X_3 is not associated with the conditions $\perp X_{\mathcal{M}_0}$, $\perp X_{\mathcal{M}_1}$, or the given density $p(x_M)$; and X_2 is not associated with the conditions $\perp X_{\mathcal{M}_0}$, $\perp X_{\mathcal{M}_1}$, $\perp X_{\mathcal{M}_3}$, or the given density $p(x_M)$.

More generally, the constraints associated with the marginalization-invariant Markov property are ordered, with respect to the set M, if we require M to contain only the leaf vertices of a rooted tree, as evidenced by the following proposition.

Proposition 2.5 (Ordered Marginalization-Invariant Constraints).
Let $\mathcal{G}_{\preceq}$ be a rooted tree, and let $(v_1, \ldots, v_m)$ be an arbitrary ordering on the non-leaf vertices of $\mathcal{G}_{\preceq}$. Suppose the marginalization constraint set M is the set of all leaf vertices of $\mathcal{G}_{\preceq}$. Then, the conditional independence constraints $\perp X_{\mathcal{M}_{v_1}}, \ldots, \perp X_{\mathcal{M}_{v_m}}$ are ordered with respect to M.

Proof. Consider any family $\mathcal{M}_{v_i}$, and note that $\cap \mathcal{M}_{v_i} = \{v_i\}$. Using (2.31) and the fact that v_i is not an element of M gives $v_i \notin M^{(i-1)} \supset \cdots \supset M^{(0)} \supset M$, and consequently, $v_i \notin M$ and $v_i \notin \cup \mathcal{M}_{v_j}$ for all $v_j < v_i$. ■

In Example 2.7, we exploited the fact that the constraints $\perp X_{\mathcal{M}_{v_i}}$ were ordered in order to design variables one at a time in a successive fashion. However, we were also concerned with ensuring consistency among the marginal densities; specifically, we found a previously defined joint density, marginalized out the relevant variables, and then defined a new joint density. In so doing, we ensure that any two densities defined in this sequential procedure agree with each other along their shared variables. For example, as shown in Figure 2.12, the constraints $\perp X_{\mathcal{M}_3}$ require an appropriate joint density $p(x_1, x_3, x_7, x_8)$ to be specified, but in this particular example, the variables X_1, X_7, and X_8 have already been defined. Consequently, to ensure consistency, we marginalize the previously defined density $p(x_0, x_1, x_7, \ldots, x_{10})$ to give $p(x_1, x_7, x_8)$, and we then use $p(x_1, x_7, x_8)$ in defining the new density $p(x_1, x_3, x_7, x_8)$.

The fact that we can always find such a previously defined density is an inherent nesting property of the families $\mathcal{M}_{v_i}$, which we now describe. In order to satisfy the constraint $\perp X_{\mathcal{M}_{v_i}}$, it is necessary to specify a density $p(x_{\cup\mathcal{M}_{v_i}})$. In our approach to sequential realization, we consider $X_{\cap\mathcal{M}_{v_i}} = X_{v_i}$ to be the design variable, and in order to maintain consistency, we need the remaining non-design variables to be defined in a previously specified density. For notational convenience, we let $N_i \triangleq \cup\mathcal{M}_{v_i} - \{v_i\}$, so that X_{N_i} represents the non-design variables. The density $p(x_{\cup\mathcal{M}_{v_i}})$ may then be written as the product $\bar{p}(x_{v_i}|x_{N_i})p(x_{N_i})$, where $\bar{p}(x_{v_i}|x_{N_i})$ is the density to be designed. In order for the sequential realization procedure to be well-defined, we need $p(x_{N_i})$ to be a marginal of a previously designed density, and this is true only if $N_i \subseteq \cup\mathcal{M}_{v_j}$ for some $v_j < v_i$. The following proposition shows that this required nesting property is true for any choice of the marginalization constraint set M.

Proposition 2.6 (Nested Marginalization-Invariant Constraints).
Let $\mathcal{G}_{\preceq}$ be a rooted tree, and let $(v_1, \ldots, v_m)$ be an arbitrary ordering on the non-leaf vertices of $\mathcal{G}_{\preceq}$. Given any marginalization constraint set $M \subseteq V$, the sets $\cup\mathcal{M}_{v_i}$ are nested in the following sense:

$$\cup\mathcal{M}_{v_1} - \{v_1\} \subseteq M \tag{2.36a}$$

$$\cup\mathcal{M}_{v_i} - \{v_i\} \subseteq \cup\mathcal{M}_{v_j} \text{ for some } v_j < v_i, \text{ and } i = 2, \ldots, m. \tag{2.36b}$$

If $v_1 \notin M$, then (2.36a) is an equality.

Proof. See Appendix A.4. ■

Having identified and discussed two important properties of the families $\mathcal{M}_{v_i}$, we are now in a position to propose a general algorithm for realizing an exact multiscale model in a sequential fashion. Subsequently, in Proposition 2.7, we show that this algorithm is well-defined if M is precisely equal to the set of all leaf vertices of the graph.

Algorithm 2.1 (Sequential Realization of Multiscale Models).
Let $\mathcal{G}_{\preceq}$ be a rooted tree, and let $(v_1, \ldots, v_m)$ be an arbitrary ordering on the non-leaf vertices of $\mathcal{G}_{\preceq}$. Suppose a density $p(x_M)$ is specified for some set $M \subseteq V$.

(1) Specify a density $\bar{p}(x_{v_1}|x_M)$ such that the conditional independence constraint $\perp X_{\mathcal{M}_{v_1}}$ is satisfied, under the joint density $p^{(1)}(x_{v_1}, x_M) \triangleq p(x_{v_1}, x_M) = \bar{p}(x_{v_1}|x_M)p(x_M)$.

(2) For $i = 2, \ldots, m$:

   (a) Define $N_i \triangleq \cup\mathcal{M}_{v_i} - \{v_i\}$, and find a vertex $v_j < v_i$ such that $N_i \subseteq \cup\mathcal{M}_{v_j}$.

   (b) Using the density $p^{(j)}(\cdot)$, marginalize away the variables indexed by $\cup\mathcal{M}_{v_j} - N_i$, to give a density $p(x_{N_i})$.

   (c) Specify a density $\bar{p}(x_{v_i}|x_{N_i})$ such that the conditional independence constraint $\perp X_{\mathcal{M}_{v_i}}$ is satisfied under the joint density $p^{(i)}(x_{v_i}, x_{N_i}) \triangleq p(x_{v_i}, x_{N_i}) = \bar{p}(x_{v_i}|x_{N_i})p(x_{N_i})$.

(3) Form the density $q(x_V) \triangleq \prod_{v\in V} p(x_v|x_{\pi(v)})$. ◀

Proposition 2.7 (Appropriateness of Algorithm 2.1).
Let $\mathcal{G}_{\preceq}$ be a rooted tree, and let $(v_1, \ldots, v_m)$ be an arbitrary ordering on the non-leaf vertices of $\mathcal{G}_{\preceq}$. Suppose $p(x_M)$ is a density specified on the set of all leaf vertices of $\mathcal{G}_{\preceq}$. Then, Algorithm 2.1 is well-defined and generates a multiscale model $(X, \mathcal{G}_{\preceq})$ with density $q(\cdot)$ satisfying $q(x_M) = p(x_M)$.

Proof. The result directly follows from Proposition 2.5, Proposition 2.6, and Theorem 2.3. ■

While the preceding proposition is encouraging, it raises two important questions about the set M:

(1) What happens when M is a proper subset of the leaf vertices of a graph?

(2) What happens when M contains some non-leaf vertices?

The first issue is a simple one to address. Suppose that M is a proper subset of the leaf vertices of a graph, so that there is at least one leaf vertex $v' \notin M$. Using (2.31) and (2.32), v' is not contained in any of the families $\mathcal{M}_{v_i}$, and consequently, it is an extraneous vertex that may be safely removed from the model without any loss to the realization problem. As a result, we henceforth assume that M contains at least all of the leaf vertices; otherwise, the tree can be "pruned" until this is the case.

The second issue is more complicated. Suppose M contains a subset of the non-leaf vertices, and let $(v_1, \ldots, v_m)$ be an arbitrary ordering on the non-leaf vertices. Then, the conditions for marginalization-invariant Markovianity become *unorderable* with respect to the set M, *i.e.* no permutation of the constraints $\perp X_{\mathcal{M}_{v_1}}, \ldots, \perp X_{\mathcal{M}_{v_m}}$ satisfies Definition 2.11. Therefore, if M contains non-leaf vertices, we cannot introduce design variables in a successive fashion, and consequently, it may not be possible to sequentially realize a multiscale model.

The fact that this particular choice of M does not allow the constraints to be ordered actually points to a related issue. By including non-leaf vertices in M, we may limit the types of densities $p(x_M)$ that can be exactly realized by a multiscale model. As an illustration, consider the graph shown in Figure 2.8, and suppose the goal is to realize a multiscale model that matches a given density $p(x_2, x_3, x_4, x_5, x_6)$. Thus, $M = \{2, 3, 4, 5, 6\}$ contains the non-leaf vertex 2. Given the ordering $(2, 1, 0)$ on the non-leaf vertices, the family $\mathcal{M}_2$ is equal to

$$\mathcal{M}_2 = \{\{2, 3, 4\}, \{2, 5\}, \{2, 6\}\},$$

and consequently, the independence condition $\perp X_{\mathcal{M}_2}$ requires $X_{\{3,4\}}$, X_5, and X_6 to be conditionally independent given $X_2 = x_2$, *i.e.*

$$p(x_2, x_3, x_4, x_5, x_6) = p(x_3, x_4|x_2)p(x_5|x_2)p(x_6|x_2). \tag{2.37}$$

Thus, for the ordering $(2, 1, 0)$ on the non-leaf vertices, the realization problem can be solved when the given density $p(x_2, x_3, x_4, x_5, x_6)$ satisfies (2.37) and therefore, has a specific conditional independence structure. If we choose a different ordering (v_1, v_2, v_3) on the non-leaf vertices, the conditions $\perp X_{\mathcal{M}_{v_1}}, \perp X_{\mathcal{M}_{v_2}}, \perp X_{\mathcal{M}_{v_3}}$ will require at least some marginal of the density $p(x_2, x_3, x_4, x_5, x_6)$, *e.g.* $p(x_2, x_5, x_6)$, to possess a conditional independence structure similar to (2.37).

In particular applications, a specified density $p(x_M)$ may have a special conditional independence structure, such as (2.37), and consequently, there are cases where Algorithm 2.1 can be used to sequentially realize a multiscale model that exactly matches such a density $p(x_M)$. For problems where $p(x_M)$ does not have such special structure, we must do something different. One possibility for dealing with this issue is to change the structure of the graph $\mathcal{G}_{\preceq}$ and/or re-map the vectors X_v, $v \in M$, such that the resulting marginalization constraint M' contains only the leaf vertices of the new graph $\mathcal{G}'_{\preceq}$. This approach is unsatisfying because as we later illustrate in Section 2.8.2, it is sometimes desirable to have M include non-leaf vertices. We focus instead on a second approach developed in the following section.

■ 2.7.3 Sequential Realization of Multiscale Models Using Augmented States

In this section, we continue our discussion of sequential realization by addressing the difficulty encountered in the previous section. We show that if M contains any non-leaf vertex v, then a sequential realization procedure is possible if the vector X_v is *augmented* with an additional design vector. Intuitively, this process of augmentation provides additional degrees of freedom in the realization problem thereby permitting a sequential realization procedure and in theory allowing any specified density $p(x_M)$ to be realized. To illustrate this idea of augmentation, we provide an example.

Example 2.8 (Sequential Realization Using Augmented States).
Our goal in providing this example is three-fold. First, we introduce the idea of an augmented state and show how to add additional design variables to the realization problem. Second, we show that a result similar to Theorem 2.3 continues to hold for a new set of families $\mathcal{M}^{\sharp}_{v_i}$, which incorporate this notion of augmentation. Finally, we show that the families $\mathcal{M}^{\sharp}_{v_i}$ satisfy the orderability and nesting properties discussed in the previous section, and consequently, it is possible to develop a sequential realization procedure.

Consider again Example 2.5 and Figure 2.8, where we imposed a marginal constraint on the leaves of the multiscale model and chose the ordering $(0, 1, 2)$ on the non-leaf vertices. Suppose that our goal is to now satisfy $p(x_2, x_3, x_4, x_5, x_6) = q(x_2, x_3, x_4, x_5, x_6)$, where $M = \{2, 3, 4, 5, 6\}$ includes the non-leaf vertex 2. As discussed in the previous section, including vertex 2 in the set M implies that only a subset of densities $p(x_2, x_3, x_4, x_5, x_6)$ may be realized with the graph structure shown in Figure 2.8. In addition, the families

$$\mathcal{M}_0 = \{\{0, 3, 4\}, \{0, 2, 5, 6\}\} \tag{2.38a}$$
$$\mathcal{M}_1 = \{\{1, 3\}, \{1, 4\}, \{1, 0\}\} \tag{2.38b}$$
$$\mathcal{M}_2 = \{\{2, 5\}, \{2, 6\}, \{2, 0\}\}, \tag{2.38c}$$

are unorderable with respect to M, and therefore, a sequential realization procedure is not possible.

To address this issue, let vector X_2 be composed of two sub-vectors $X_{2^{(d)}}$ and $X_{2^{(t)}}$, *i.e.* $X_2 = \{X_{2^{(d)}}, X_{2^{(t)}}\}$, and consider a new marginalization constraint set $M^{\sharp} = \{3, 4, 5, 6, 2^{(t)}\}$.[18] Our goal in this example is to design a multiscale model $q(\cdot)$ which satisfies the marginal constraint $q(x_{M^{\sharp}}) = p(x_{M^{\sharp}})$ for a specified density $p(x_{M^{\sharp}})$. By splitting X_2 into two sub-vectors, we have additional degrees of freedom which we did not previously have – this is the concept of augmentation previously mentioned. The sub-vector $X_{2^{(d)}}$ implicitly represents these additional degrees of freedom, serving as a design vector in the realization problem, and the sub-vector $X_{2^{(t)}}$ is what we call a *target* vector since it is part of the specified density $p(x_{M^{\sharp}})$.

[18] This particular notation is defined later in this section.

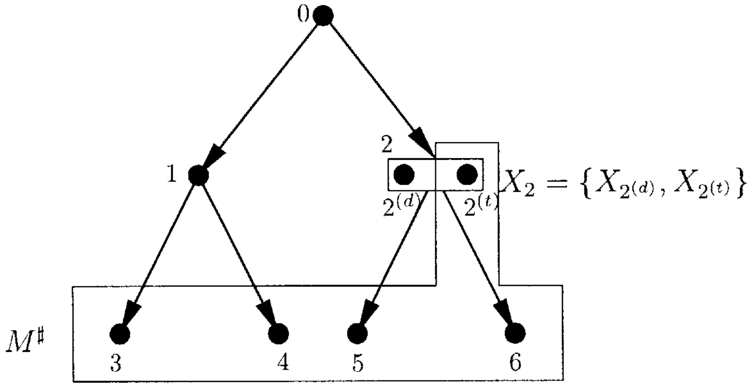


Figure 2.13. A graphical representation of the state augmentation problem where vertex 2 is split into two separate vertices $2^{(d)}$ and $2^{(t)}$, and $X_2 = \{X_{2^{(d)}}, X_{2^{(t)}}\}$ is composed of both a design vector $X_{2^{(d)}}$ and a target vector $X_{2^{(t)}}$. The marginalization constraint set $M^\sharp$ contains the vertices 3, 4, 5, 6, and $2^{(t)}$.

In order to incorporate the notion of augmentation into our set-theoretic notation, we introduce the artifice of a "split" vertex. That is, we now consider vertex 2 to be a tuple of vertices, *i.e.* $2 = (2^{(d)}, 2^{(t)})$, where 2, $2^{(d)}$, and $2^{(t)}$ all have the same physical location in the graph $\mathcal{G}_{\preceq}$ but simply index different parts of the process X. Figure 2.13 graphically illustrates this idea.

Having introduced an additional design variable into the realization problem, we now consider a new set of families $\mathcal{M}^\sharp_{v_i}$ as follows,

$$\mathcal{M}^\sharp_0 = \{\{0,3,4\},\{0,2^{(t)},5,6\}\} \tag{2.39a}$$

$$\mathcal{M}^\sharp_1 = \{\{1,3\},\{1,4\},\{1,0\}\} \tag{2.39b}$$

$$\mathcal{M}^\sharp_2 = \{\{2,5\},\{2,6\},\{2,0\}\} = \{\{2^{(d)},2^{(t)},5\},\{2^{(d)},2^{(t)},6\},\{2^{(d)},2^{(t)},0\}\}. \tag{2.39c}$$

Notice that the families $\mathcal{M}^\sharp_{v_i}$ are identical to those in (2.38) except that vertex 2 in $\mathcal{M}_0$ has been replaced by vertex $2^{(t)}$ in $\mathcal{M}^\sharp_0$.[19] Since the families $\mathcal{M}^\sharp_{v_i}$ and $\mathcal{M}_{v_i}$ are different, we need a modified version of Theorem 2.3 that accounts for the separation of design and target vectors. To proceed, we must show that a density $p(\cdot)$ satisfying $\perp X_{\mathcal{M}^\sharp_0}$, $\perp X_{\mathcal{M}^\sharp_1}$, and $\perp X_{\mathcal{M}^\sharp_2}$ does in fact generate a density $q(x) = \prod_{v \in V} p(x_v | x_{\pi(v)})$, with $q(x_{M^\sharp}) = p(x_{M^\sharp})$.

The method of proof is the same as that given in Example 2.5. We begin by imposing the two constraints $\mathcal{M}^\sharp_2$ and $\mathcal{M}^\sharp_1$ on $p(\cdot)$, and since this was already performed in Example 2.5, we borrow the result from (2.29),

$$q(x) = p(x_0)p(x_3,x_4|x_0)p(x_5,x_6|x_0)p(x_2|x_0,x_5,x_6)p(x_1|x_0,x_3,x_4).$$

Splitting variable X_2 into its two components $X_{2^{(d)}}$ and $X_{2^{(t)}}$ and manipulating the densities gives

[19] We use the notation $\mathcal{M}^\sharp_{v_i}$ to remind the reader that these are not the usual families $\mathcal{M}_{v_i}$. Later in this section, we provide a rule for obtaining these types of families.

the following,

$$\begin{aligned} q(x) &= p(x_0)p(x_3, x_4|x_0)p(x_5, x_6|x_0)p(x_{2^{(d)}}, x_{2^{(t)}}|x_0, x_5, x_6)p(x_1|x_0, x_3, x_4) \\ &= p(x_0)p(x_3, x_4|x_0)p(x_5, x_6|x_0)p(x_{2^{(t)}}|x_0, x_5, x_6)p(x_{2^{(d)}}|x_0, x_5, x_6, x_{2^{(t)}})p(x_1|x_0, x_3, x_4) \\ &= p(x_0)p(x_3, x_4|x_0)p(x_5, x_6, x_{2^{(t)}}|x_0)p(x_{2^{(d)}}|x_0, x_5, x_6, x_{2^{(t)}})p(x_1|x_0, x_3, x_4). \end{aligned} \tag{2.40}$$

Imposing the constraint $\perp X_{\mathcal{M}_0^\sharp}$ on $p(\cdot)$, which requires

$$p(x_0, x_3, x_4, x_5, x_6, x_{2^{(t)}}) = p(x_0)p(x_3, x_4|x_0)p(x_5, x_6, x_{2^{(t)}}|x_0),$$

gives

$$q(x) = p(x_0, x_3, x_4, x_5, x_6, x_{2^{(t)}})p(x_{2^{(d)}}|x_0, x_5, x_6, x_{2^{(t)}})p(x_1|x_0, x_3, x_4), \tag{2.41}$$

and integrating out the variables $x_{2^{(d)}}$, x_1, and x_0 (in that order) shows that $q(x_3, x_4, x_5, x_6, x_{2^{(t)}}) = p(x_3, x_4, x_5, x_6, x_{2^{(t)}})$.

Having proven the preceding fact, we can now proceed to designing a sequential realization procedure, but this task is simple given the discussion in the previous section. Notice that the constraints $\perp X_{\mathcal{M}_0^\sharp}$, $\perp X_{\mathcal{M}_1^\sharp}$, and $\perp X_{\mathcal{M}_2^\sharp}$ are ordered with respect to $M^\sharp = \{3, 4, 5, 6, 2^{(t)}\}$, thereby ensuring that the design variables X_0, X_1, and $X_{2^{(d)}}$ may be introduced in a successive fashion. In addition, the families $\mathcal{M}_0^\sharp$, $\mathcal{M}_1^\sharp$, and $\mathcal{M}_2^\sharp$ exhibit a nesting property similar to the one previously discussed in Proposition 2.6, thereby allowing us to maintain consistency amongst the marginals.[20] Figure 2.14 shows the resulting sequential realization procedure for this example. ◀

In the remainder of this section, we proceed in a manner similar to that provided in Example 2.8. First, we introduce the necessary notation and provide a rule for obtaining the families $\mathcal{M}_{v_i}^\sharp$. Second, we provide a strengthened version of Theorem 2.3 that applies to the families $\mathcal{M}_{v_i}^\sharp$. Finally, we provide results similar to those given in the previous section, ultimately arriving at a general algorithm for sequential realization.

To accommodate augmented states, we now allow every vertex $v \in V$ to have two labels $v^{(d)}$ and $v^{(t)}$; these labels provide a convenient means of indexing relevant parts of the process X. In particular, for each vertex $v \in V$, we have the following three types of vectors:

(1) $X_{v^{(d)}}$ – refers to the design vector at vertex v

(2) $X_{v^{(t)}}$ – refers to the target vector at vertex v

(3) $X_v = \{X_{v^{(d)}}, X_{v^{(t)}}\}$ – refers to the entire vector at vertex v.

Of course, for some vertices v, the two vectors $X_{v^{(d)}}$ and X_v are identical; this occurs when v is a non-leaf vertex containing no target vector, *i.e.* v is not part of the constraint set M. Alternatively, for leaf vertices v, the two vectors $X_{v^{(t)}}$ and X_v are identical because a design vector at a leaf vertex plays no role in the realization problem.

Suppose $\mathcal{G}_\preceq = (V, E)$ is a rooted tree, $M \subseteq V$ is a marginalization constraint set, and $(v_1, \ldots, v_m)$ is an ordering on the non-leaf vertices. Using the families $\mathcal{M}_{v_i}$ characterized in (2.32), we now define a set of *augmented marginalization-invariant families* $\mathcal{M}_{v_i}^\sharp$ and an *augmented marginalization constraint set* $M^\sharp$ via the following augmentation rule.

[20] We provide the details of this nesting property later in Proposition 2.8.

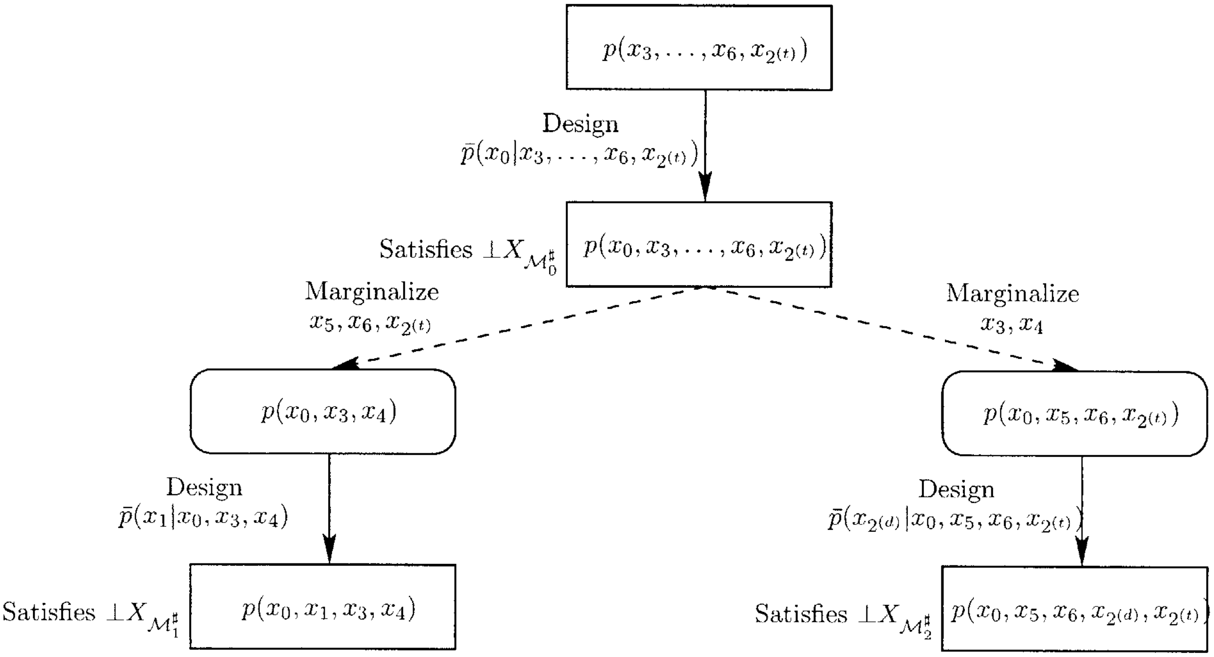

$p(x_3, \ldots, x_6, x_{2^{(t)}})$
Design
$\bar{p}(x_0|x_3, \ldots, x_6, x_{2^{(t)}})$
Satisfies $\perp X_{\mathcal{M}_0^\sharp}$
$p(x_0, x_3, \ldots, x_6, x_{2^{(t)}})$
Marginalize
$x_5, x_6, x_{2^{(t)}}$
Marginalize
x_3, x_4
$p(x_0, x_3, x_4)$
$p(x_0, x_5, x_6, x_{2^{(t)}})$
Design
$\bar{p}(x_1|x_0, x_3, x_4)$
Design
$\bar{p}(x_{2^{(d)}}|x_0, x_5, x_6, x_{2^{(t)}})$
Satisfies $\perp X_{\mathcal{M}_1^\sharp}$
$p(x_0, x_1, x_3, x_4)$
Satisfies $\perp X_{\mathcal{M}_2^\sharp}$
$p(x_0, x_5, x_6, x_{2^{(d)}}, x_{2^{(t)}})$

Figure 2.14. Block diagram illustrating the steps involved in a sequential realization of the multiscale model considered in Example 2.8 and Figure 2.13, with the ordering $(0, 1, 2)$ on the non-leaf vertices and $M^\sharp = \{3, 4, 5, 6, 2^{(t)}\}$. The rectangular boxes contain the densities needed to satisfy the conditions $\perp X_{\mathcal{M}_{v_i}^\sharp}$, while the rounded boxes contain intermediate densities which result from a marginalization step. The dashed arrows indicate a marginalization of a density, while the solid arrows represent a design step where a conditional density must be specified.

Augmentation Rule:

(0) Let $M^\sharp = M$ and $\mathcal{M}_{v_i}^\sharp = \mathcal{M}_{v_i}$ for $i = 1, \ldots, m$.

(1) Replace each vertex $v \in M^\sharp$ with $v^{(t)}$.

(2) For each $v_i \in (v_1, \ldots, v_m)$ and each $v \in \cup\mathcal{M}_{v_i}^\sharp$ do the following:

   (i) If v is a leaf vertex or if v is a non-leaf vertex with $v > v_i$, replace v with $v^{(t)}$.

   (ii) If v is a non-leaf vertex with $v \leq v_i$ and $v \in M$, replace v with the tuple $v^{(d)}, v^{(t)}$.

   (iii) If v is a non-leaf vertex with $v \leq v_i$ and $v \notin M$, replace v with $v^{(d)}$.

Given a specified density $p(x_M)$, step (1) in the preceding rule creates an equivalent density $p(x_{M^\sharp})$ which uses the new indexing scheme introduced in this section. The consequence of step (2) is that for any $v_i \in (v_1, \ldots, v_m)$, the constraint $\perp X_{\mathcal{M}_{v_i}^\sharp}$ considers the entire vector X_v for vertices $v \leq v_i$ and considers only the partial vector $X_{v^{(t)}}$ for vertices $v > v_i$.[21] These steps generate a set

[21] In this rule, we have chosen to be explicit about what part of the vector X_v is included in each constraint $\perp X_{\mathcal{M}_{v_i}^\sharp}$. For example, if $v \notin M$, then there is no target vector $X_{v^{(t)}}$ (*i.e.* $X_v = X_{v^{(d)}}$), and consequently, we do not include the vertex $v^{(t)}$ in $\mathcal{M}_{v_i}^\sharp$. This explicitness is necessary in order for the families $\mathcal{M}_{v_i}^\sharp$ to possess the nesting property provided in Proposition 2.8.

of families $\mathcal{M}^\sharp_{v_1}, \ldots, \mathcal{M}^\sharp_{v_m}$ and corresponding independence conditions $\perp X_{\mathcal{M}^\sharp_{v_1}}, \ldots \perp X_{\mathcal{M}^\sharp_{v_m}}$ which, as we later indicate, permit a sequential realization procedure.

The augmentation rule allows us to reformulate Theorem 2.3 for the more general problem of interest here.

Theorem 2.4 (Marginalization-Invariance for Augmented States).
Suppose random vectors $\{X_v\}_{v \in V}$ admit a probability density $p(\cdot)$; define $q(x) \triangleq \prod_{v \in V} p(x_v | x_{\pi(v)})$. In addition, suppose the families $\mathcal{M}^\sharp_{v_1}, \ldots, \mathcal{M}^\sharp_{v_m}$ satisfy the augmentation rule for some specified set M and an ordering $(v_1, \ldots, v_m)$ on the non-leaf vertices of $\mathcal{G}_\preceq$. If $\{X_v\}$ satisfies the conditional independence constraints $\perp X_{\mathcal{M}^\sharp_{v_1}}, \ldots, \perp X_{\mathcal{M}^\sharp_{v_m}}$ then $q(x_{M^\sharp}) = p(x_{M^\sharp})$.

Proof. See Section 3.9.2. ■

Given Theorem 2.4, it is now possible to propose a sequential realization procedure using augmented states. The intuition behind this procedure is the same as that given previously in Section 2.7.2, and therefore, we simply state the relevant results.

Proposition 2.8 (Properties of Augmented Marginalization-Invariant Families).
Let $\mathcal{G}_\preceq$ be a rooted tree, and let $(v_1, \ldots, v_m)$ be an arbitrary ordering on the non-leaf vertices of $\mathcal{G}_\preceq$. Given <u>any</u> marginalization constraint set $M \subseteq V$, the following is true:

(1) The constraints $\perp X_{\mathcal{M}^\sharp_{v_1}}, \ldots, \perp X_{\mathcal{M}^\sharp_{v_m}}$ are ordered with respect to $M^\sharp$.

(2) The sets $\cup \mathcal{M}^\sharp_{v_i}$ are nested in the following sense:

$$\cup \mathcal{M}^\sharp_{v_1} - \{v_1^{(d)}\} = M^\sharp \tag{2.42a}$$

$$\cup \mathcal{M}^\sharp_{v_i} - \{v_i^{(d)}\} \subseteq \cup \mathcal{M}^\sharp_{v_j} \text{ for some } v_j < v_i, \text{ and } i = 2, \ldots, m. \tag{2.42b}$$

Proof. See Appendix A.4. ■

Algorithm 2.2 (Sequential Realization Using Augmented States).
Let $\mathcal{G}_\preceq$ be a rooted tree, and let $(v_1, \ldots, v_m)$ be an arbitrary ordering on the non-leaf vertices of $\mathcal{G}_\preceq$. Suppose a density $p(x_M)$ is specified for some set $M \subseteq V$.

(1) Specify a density $\bar{p}(x_{v_1^{(d)}} | x_{M^\sharp})$ such that the conditional independence constraint $\perp X_{\mathcal{M}^\sharp_{v_1}}$ is satisfied, under the joint density $p^{(1)}(x_{v_1^{(d)}}, x_{M^\sharp}) \triangleq p(x_{v_1^{(d)}}, x_{M^\sharp}) = \bar{p}(x_{v_1^{(d)}} | x_{M^\sharp}) p(x_{M^\sharp})$.

(2) For $i = 2, \ldots, m$:

  (a) Define $N_i \triangleq \cup \mathcal{M}^\sharp_{v_i} - \{v_i^{(d)}\}$, and find a vertex $v_j < v_i$ such that $N_i \subseteq \cup \mathcal{M}^\sharp_{v_j}$.

  (b) Using the density $p^{(j)}(\cdot)$, marginalize away the variables indexed by $\cup \mathcal{M}^\sharp_{v_j} - N_i$, to give a density $p(x_{N_i})$.

  (c) Specify a density $\bar{p}(x_{v_i^{(d)}} | x_{N_i})$ such that the conditional independence constraint $\perp X_{\mathcal{M}^\sharp_{v_i}}$ is satisfied under the joint density $p^{(i)}(x_{v_i^{(d)}}, x_{N_i}) \triangleq p(x_{v_i^{(d)}}, x_{N_i}) = \bar{p}(x_{v_i^{(d)}} | x_{N_i}) p(x_{N_i})$.

(3) Form the density $q(x_V) \triangleq \prod_{v \in V} p(x_v | x_{\pi(v)})$. ◀

Proposition 2.9 (Significance of Algorithm 2.2).

Let $\mathcal{G}_{\preceq}$ be a rooted tree, and let $(v_1, \ldots, v_m)$ be an arbitrary ordering on the non-leaf vertices of $\mathcal{G}_{\preceq}$. Suppose $p(x_M)$ is a density specified on a set of the vertices $M \subseteq V$. Then, Algorithm 2.2 is well-defined and generates a multiscale model $(X, \mathcal{G}_{\preceq})$ with density $q(\cdot)$ satisfying $q(x_{M^\sharp}) = p(x_{M^\sharp})$.

Proof. The result directly follows from Proposition 2.8 and Theorem 2.4. ■

In summary, we have shown that any realization problem may in theory be solved sequentially if enough design variables are introduced. In practice, the design steps required in Algorithms 2.1 and 2.2, namely step 1 and step 2(c), can be very challenging. The next section includes several academic examples where each design step can be solved in a simple manner, and in subsequent chapters, we develop methods for approximating the design steps for more complex applications.

■ 2.8 Ties To Earlier Work

We conclude this chapter with a series of examples illustrating the usefulness of the methodology described in the preceding sections. In what follows, we do not provide new algorithms for realizing multiscale models, but instead, we offer a new perspective on existing results by showing how several important topics in multiscale realization theory fit into our framework.

■ 2.8.1 Markov Processes

The class of Markov processes has served as a particularly important example throughout the development of multiscale realization theory. The work of [75] showed that multiscale models can be constructed to match the statistics of these types of processes exactly, and in several different applications, estimation tasks have been performed with respect to a multiscale prior model of a Markov process [47, 73]. This example shows how Theorem 2.3 may be used to construct a multiscale model for a Markov process mapped to the leaf vertices of a tree.

Consider the 16 point first-order Markov process $Y = \{Y_i\}_{1 \le i \le 16}$ graphically represented in Figure 2.15(a). Further, suppose that the goal is to construct a multiscale model with the graph structure shown in Figure 2.15(b) and where the process Y is mapped to the leaf vertices of the graph. In order to realize the process Y, Theorem 2.3 indicates that the constraints $\perp X_{\mathcal{M}_{v_i}}$ must be satisfied (here we arbitrarily chose the ordering $(0, 1, 2)$) for the following families:

$$\begin{aligned}
\mathcal{M}_0 &= \{\{0,3,4\},\{0,5,6\}\} \\
\mathcal{M}_1 &= \{\{1,3\},\{1,4\},\{0,1\}\} \\
\mathcal{M}_2 &= \{\{2,5\},\{2,6\},\{0,2\}\}.
\end{aligned}$$

Proceeding in a sequential fashion, as described earlier in Section 2.7.1, we first design the vector X_0 so that $X_{\{3,4\}}$ and $X_{\{5,6\}}$ are conditionally independent. Using the properties of the first-order Markov process, note that either $X_0 = \{Y_8\}$ or $X_0 = \{Y_9\}$ is a valid choice. To satisfy $\perp X_{\mathcal{M}_1}$, we must design X_1 to make X_3,X_4, and X_0 conditionally independent, and choosing either $X_1 = \{Y_4, Y_8\}$ or $X_1 = \{Y_5, Y_8\}$ satisfies this goal for both choices of X_0. Finally, to satisfy $\perp X_{\mathcal{M}_2}$ and make X_5, X_6, and X_0 conditionally independent given the value of X_2, we can either choose $X_2 = \{Y_9, Y_{12}\}$ or $X_2 = \{Y_9, Y_{13}\}$.

Notice that this sequential realization procedure leads to eight different multiscale models, each satisfying the required marginal constraint; Figure 2.15(c) graphically displays one of these models.

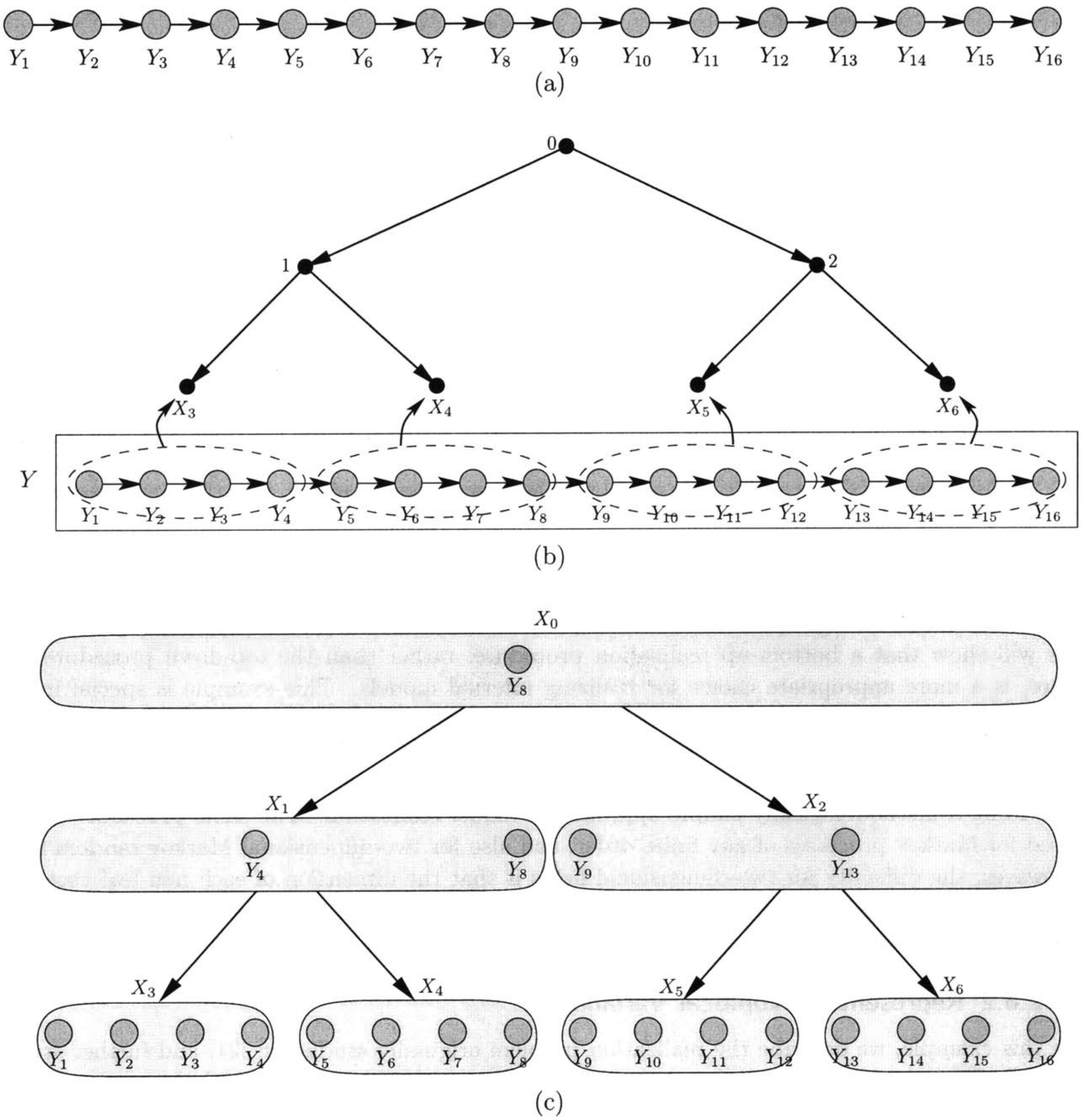

Y_1 Y_2 Y_3 Y_4 Y_5 Y_6 Y_7 Y_8 Y_9 Y_{10} Y_{11} Y_{12} Y_{13} Y_{14} Y_{15} Y_{16}
(a)
0
1
2
X_3
X_4
X_5
X_6
Y
Y_1 Y_2 Y_3 Y_4 Y_5 Y_6 Y_7 Y_8 Y_9 Y_{10} Y_{11} Y_{12} Y_{13} Y_{14} Y_{15} Y_{16}
(b)
X_0
Y_8
X_1
X_2
Y_4
Y_8
Y_9
Y_{13}
X_3
X_4
X_5
X_6
Y_1 Y_2 Y_3 Y_4 Y_5 Y_6 Y_7 Y_8 Y_9 Y_{10} Y_{11} Y_{12} Y_{13} Y_{14} Y_{15} Y_{16}
(c)

Figure 2.15. (a) A 16 point first-order Markov process, Y. (b) Mapping of the process Y to the leaf vertices of rooted tree. (c) One possible multiscale model that exactly realizes the statistics of Y.

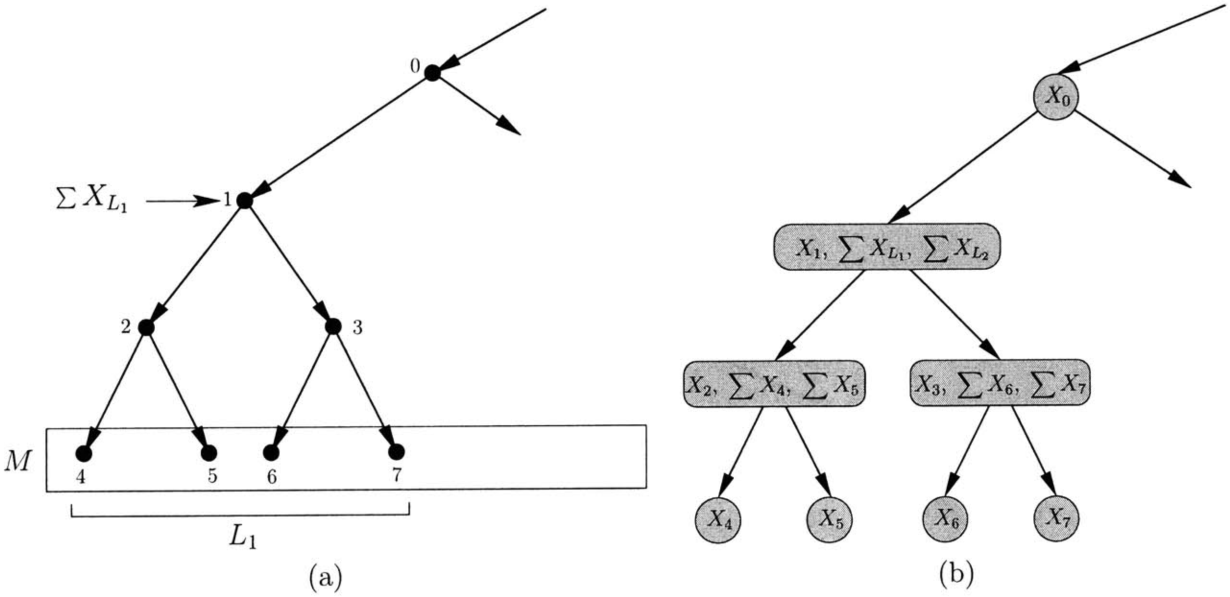

0
ΣX_{L_1}
1
2
3
M
4
5
6
7
L_1
(a)
X_0
$X_1, \sum X_{L_1}, \sum X_{L_2}$
$X_2, \sum X_4, \sum X_5$
$X_3, \sum X_6, \sum X_7$
X_4
X_5
X_6
X_7
(b)

Figure 2.16. (a) An example of a state-augmentation problem where the value of $\sum X_{L_1}$ must be contained at vertex 1 and the finest-scale statistics must match some specified density $p(x_M)$. (b) Assuming $(X, \mathcal{G}_{\preceq})$ is a multiscale model with finest-scale density $p(x_M)$, then the figure illustrates one possible solution to the realization problem considered in (a).

Notice also that all eight possibilities correspond to internal multiscale models. In Section 2.8.3, we will show that a bottom-up realization procedure, rather than the top-down procedure used here, is a more appropriate choice for realizing internal models. This example is special in that the particular ordering on the non-leaf vertices is not important; the reader can verify that any ordering yields the same eight possibilities.

This example, while somewhat trivial in nature, demonstrates the utility of Theorem 2.3 in providing a methodologically simple approach to model realization. The same procedure may be used for Markov processes of any finite order, and also for two-dimensional Markov random fields. However, the difficulty for two-dimensional fields is that the dimension of each non-leaf vector X_v can become large when designing exact models. One approach to dealing with this problem is to design approximate multiscale models as discussed in Chapter 3.

■ 2.8.2 Representing Nonlocal Variables

In this example, we describe the realization problem originally studied in [21] and further studied in [38]. The goal is to specify a multiscale model that exactly represents: (1) the statistics of an observed finest-scale process and (2) some specified linear functions of the finest-scale process. We call these additional linear functions, *nonlocal variables.* For example, consider the graph shown in Figure 2.16(a), where a process X_M is mapped to the leaf vertices. We want to design a model that exactly captures the statistics of X_M, and in addition, we require vertex 1 to contain the sum of the process X_{L_1}.[22]

In practice, this type of realization problem is important in two different settings:

[22] For an n-dimensional vector $X = (X_1, X_2, \ldots, X_n)$, the notation $\sum X$ is used to represent $\sum_{i=1}^{N} X_i$.

(1) In some settings, such as remote sensing [23] or geophysical applications, observed data may be available at different resolutions. In order to incorporate all such data into a single statistical model, it is necessary to "fuse" observations in a coherent manner. Multiscale models provide an effective means to perform data fusion if we map the finest resolution data to the leaves of a tree and coarser-resolution data to vertices closer to the root. In Figure 2.16(a), for example, the value of $\sum X_{L_1}$ might represent an additional observation contained within a given set of data.

(2) In other settings (see [25] for example), nonlocal variables might be introduced into a multiscale model in order to simplify an estimation task. For example, in Figure 2.16(a), the value of $\sum X_{L_1}$ can be efficiently estimated within the multiscale framework since the multiscale inference algorithm computes the best estimate of $\sum X_{L_1}$ at vertex 1.

While the example shown in Figure 2.16(a) offers a simplified perspective of the more general problem considered in [21], it emphasizes the dual role that X_1 must play. First, X_1 must satisfy the constraints placed on it by the conditional independence requirements of a multiscale model, and second, X_1 must contain a specified function of the realized finest-scale process. We describe two different approaches for solving this type of realization problem.

State Augmentation

The work of [21] showed how to augment the states of an existing multiscale model with specific nonlocal variables, without destroying the Markov properties of the model. Using the example in Figure 2.16(a), we show how the approach of [21] can be deduced directly from the marginalization-invariant Markov property. While we focus on a simple example, the ideas presented here hold more generally for adding an arbitrary number of nonlocal variables, and we refer the reader to [21] for more details.

Suppose a multiscale model $(Z, \mathcal{G}_{\preceq})$ with density p and the graph structure shown in Figure 2.16(a) is given;[23] consequently, each non-leaf vector Z_v satisfies the global Markov property. We want to realize a new multiscale model $(X, \mathcal{G}_{\preceq})$ with density q, containing $\sum Z_{L_1}$ at vertex 1 and such that the density at the finest-scale remains unchanged. It is useful to consider the sequence of steps involved in the realization problem:

(0) We are given a tree-indexed process $(Z, \mathcal{G}_{\preceq})$ characterized by a density p which recursively factors according to $\mathcal{G}_{\preceq}$.

(1) We then must create a new tree-indexed process $(X, \mathcal{G}_{\preceq})$ characterized by density $\tilde{p}(\cdot)$ such that $\tilde{p}(x_M) = p(x_M)$ and such that $\sum Z_{L_1}$ is represented at vertex 1.

(2) We then form a multiscale model $(X, \mathcal{G}_{\preceq})$ with density q, using the density factorization $q(x) = \prod_{v \in V} \tilde{p}(x_v | x_{\pi(v)})$.

Our discussion in Section 2.7.3 on sequential realization using augmented states applies to this type of problem. In particular, Theorem 2.4 specifies the requirements on the tree-indexed process

[23] This model could come from a previous realization procedure. For our purposes, we simply assume that a multiscale model is given. We use random vector Z rather than X in order to remind the reader that this is the multiscale model associated with density p.

in (1) in order for the multiscale model in (2) to satisfy the marginalization constraint $q(x_{M^\sharp}) = \tilde{p}(x_{M^\sharp}) = p(x_M, \sum x_{L_1})$, where $M^\sharp = M \cup \{1^{(t)}\}$.[24]

To identify a solution to this realization problem, we use Theorem 2.4 and Algorithm 2.2. As we show, the design steps required in Algorithm 2.2 can be solved in a simple manner because we have an existing multiscale model $(Z, \mathcal{G}_{\preceq})$ that satisfies many of the conditional independencies required by Theorem 2.4. Therefore, this type of realization problem has more structure than the generic problem considered in Section 2.7.3 because, as we show, the realization problem can be solved by making simple modifications to the existing model $(Z, \mathcal{G}_{\preceq})$. As a first step, we know that the tree-indexed process $(X, \mathcal{G}_{\preceq})$ with density $\tilde{p}(\cdot)$ must satisfy $\tilde{p}(x_{M^\sharp}) = p(x_M, \sum x_{L_1})$, as required in (1) above. To ensure that this is the case, we set $X_v = Z_v$ for all leaf vertices $v \in V$, and we set $X_{1^{(t)}} = \sum Z_{L_1}$.

Consider now any ordering of the form $(1, 2, 3, \ldots)$. For this particular ordering, the constraint $\perp X_{\mathcal{M}_1^\sharp}$ requires $X_{L_2} = Z_{L_2}$, $X_{L_3} = Z_{L_3}$, and the rest of the finest-scale process to be conditionally independent given $X_1 = \{X_{1^{(t)}}, X_{1^{(d)}}\} = \{\sum Z_{L_1}, X_{1^{(d)}}\}$. If X_1 did not contain $\sum Z_{L_1}$, then we could simply choose $X_1 = Z_1$, since the global Markov property is satisfied at vertex 1 in the existing model. Notice that the choice $X_1 = \{\sum Z_{L_1}, Z_1\}$ is not a valid solution because conditioning on $\sum Z_{L_1} = \sum z_{L_1}$ and $Z_1 = z_1$ together causes $X_{L_2} = Z_{L_2}$ and $X_{L_3} = Z_{L_3}$ to be conditionally dependent. To satisfy our objective, we must further augment the state with either $\sum Z_{L_2}$ or $\sum Z_{L_3}$ as follows,

$$X_1 = \left\{\sum Z_{L_1}, Z_1, \sum Z_{L_2}\right\} \quad \text{or} \tag{2.43a}$$

$$X_1 = \left\{\sum Z_{L_1}, Z_1, \sum Z_{L_3}\right\}. \tag{2.43b}$$

In a similar manner, the conditions $\perp X_{\mathcal{M}_2^\sharp}$ and $\perp X_{\mathcal{M}_3^\sharp}$ must be satisfied. In accordance with the condition $\perp X_{\mathcal{M}_2^\sharp}$, we design X_2 so that $X_4 = Z_4$, $X_5 = Z_5$, and X_1 are conditionally independent, and this can be accomplished, regardless of our choice in (2.43), by the following

$$X_2 = \left\{Z_2, \sum Z_4, \sum Z_5\right\}. \tag{2.44}$$

Similarly, the choice

$$X_3 = \left\{Z_3, \sum Z_6, \sum Z_7\right\} \tag{2.45}$$

makes $X_6 = Z_6$, $X_7 = Z_7$, and X_1 conditionally independent, as required by $\perp X_{\mathcal{M}_3^\sharp}$. Finally, for any non-leaf vertex $v \neq 1, 2, 3$, the choice $X_v = Z_v$ satisfies the requirements of the global Markov property and hence the conditions $\perp X_{\mathcal{M}_v^\sharp}$. Figure 2.16(b) graphically depicts one of the two solutions derived here for this simple realization problem.

In the preceding discussion, a valid solution to the realization problem was found when vertices $1, 2, 3$ were required to be first in the ordering, but this requirement was simply for clarity of exposition. In fact, the same solution satisfies the conditions $\perp X_{\mathcal{M}_{v_i}^\sharp}$ for an arbitrary ordering. Furthermore, the solution outlined here is equivalent to the state augmentation approach presented in [21]. We note also that this solution has in the past been interpreted as the one needed to maintain *consistency* or *internality* [21, 38], but from our perspective, it is in fact the solution needed to ensure that the marginal constraint is satisfied.

[24] Recall that $1^{(t)}$ is simply an index for the observed vector at vertex 1; in this case, $X_{1^{(t)}} = \sum Z_{L_1}$.

Exact Nonlocal Method

The work of [38] presents an alternative method for realizing multiscale models that contain nonlocal functions. The idea is similar in spirit to state augmentation, but this approach is different in that the nonlocal functions are mapped to the tree first and then a multiscale model is designed around the functions. From a realization standpoint, this makes more sense because it allows vectors X_v to satisfy their conditional independence roles with specific knowledge of the nonlocal functions to be represented, rather than adding nonlocal functions to an existing model and thereby increasing the dimension of each state.

The relationship between this approach to modeling and the framework discussed in this chapter is that they are identical points of view. The marginalization constraint $q(x_M) = p(x_M)$ contains all of the necessary requirements, including those required for representing nonlocal functions. We note that the algorithm presented in [38] includes an additional step which ensures that the resulting model is internal.

■ 2.8.3 Internal Models and a Scale-Recursive Algorithm

The work of [38–40] provides a computationally efficient and non-iterative algorithm for realizing Gaussian internal multiscale models where the process of interest is mapped to the leaf vertices of the graph. While this algorithm is well-suited to realize approximate models – the focus of the next chapter – we choose to discuss here the basic structure of the algorithm, and we show how this structure is similar to the methodology presented in this chapter.

First, recall that a multiscale model is internal if and only if it is locally internal, as shown in Proposition 2.1. Second, consider any bottom-up ordering on the non-leaf vertices of a graph, and recall that this implies that a vertex v cannot precede any of its children in the ordering. These two facts together suggest a sequential realization procedure which uses a bottom-up ordering to recursively define vectors X_v via local functions of already defined vectors $X_{\chi(v)}$.

In [38], this type of bottom-up procedure is used to realize a locally internal and hence internal multiscale model. However, a very specific bottom-up ordering $(v_1, \dots, v_m)$ is chosen, one that satisfies $m(v_i) \geq m(v_j)$ for all $v_i < v_j$. This implies that the ordering must start with the vertices of finest scale, then the vertices of the next finest scale, and so forth. We call this type of ordering *scale recursive*. In Figure 2.17, $(3, 4, 5, 6, 1, 2, 0)$ is an example of a scale-recursive ordering.

We now use the graph in Figure 2.17 to illustrate the relationship between marginalization-invariant Markovianity, Theorem 2.3, and the scale-recursive approach in [38]. Suppose the goal is to realize a process mapped to the finest-scale vertices of the graph in Figure 2.17, so that $M = \{7, \dots, 14\}$, and suppose the scale-recursive ordering $(3, 4, 5, 6, 1, 2, 0)$ is chosen. The first four families $\mathcal{M}_{v_i}$ are given by[25]

$$\mathcal{M}_3 = \{\{3,7\}, \{3,8\}, \{3,9,10,11,12,13,14\}\} \tag{2.46a}$$

$$\mathcal{M}_4 = \{\{4,9\}, \{4,10\}, \{4,3,11,12,13,14\}\} \tag{2.46b}$$

$$\mathcal{M}_5 = \{\{5,11\}, \{5,12\}, \{5,3,4,13,14\}\} \tag{2.46c}$$

$$\mathcal{M}_6 = \{\{6,13\}, \{6,14\}, \{6,3,4,5\}\}, \tag{2.46d}$$

and these families are graphically depicted in Figures 2.17(a),(c),(e), and (g).

[25] Recall that Proposition 2.4 characterized the families $\mathcal{M}_{v_i}$ for any bottom-up ordering when M is equal to all of the leaf vertices.

Now, consider a different set of families $\mathcal{M}'_{v_i}$ defined as follows

$$\mathcal{M}'_3 = \{\{3,7\},\{3,8\},\{3,9,10,11,12,13,14\}\} \tag{2.47a}$$
$$\mathcal{M}'_4 = \{\{4,9\},\{4,10\},\{4,7,8,11,12,13,14\}\} \tag{2.47b}$$
$$\mathcal{M}'_5 = \{\{5,11\},\{5,12\},\{5,7,8,9,10,13,14\}\} \tag{2.47c}$$
$$\mathcal{M}'_6 = \{\{6,13\},\{6,14\},\{6,7,8,9,10,11,12\}\}, \tag{2.47d}$$

and graphically depicted in Figures 2.17(b),(d),(f), and (h). Recall that our goal is to realize an internal model, and this in turn implies that X_3 is a function of $X_{\{7,8\}}$, X_4 is a function of $X_{\{9,10\}}$, and so forth. The consequence of internality is that the conditions $\perp X_{\mathcal{M}'_{v_i}}$ imply the conditions $\perp X_{\mathcal{M}_{v_i}}$, *i.e.* if $\perp X_{\mathcal{M}'_{v_i}}$ holds then $\perp X_{\mathcal{M}_{v_i}}$ holds as well. To see this, compare the left and right columns in Figure 2.17. For example, comparing Figures 2.17(d) and (c), if X_9, X_{10}, and $X_{\{7,8,11,12,13,14\}}$ are conditionally independent given $X_4 = x_4$ (or equivalently $\perp X_{\mathcal{M}'_4}$) then X_9, X_{10}, and $X_{\{3,11,12,13,14\}}$ are conditionally independent given $X_4 = x_4$ (or equivalently $\perp X_{\mathcal{M}_4}$) because X_3 is a function of X_7 and X_8. This line of reasoning indicates that an exact internal multiscale model may be realized by satisfying the conditions $\perp X_{\mathcal{M}'_{v_i}}$ (assuming $\mathcal{M}'_1$,$\mathcal{M}'_2$, and $\mathcal{M}'_0$ are defined appropriately).

The families $\mathcal{M}'_{v_i}$ defined in (2.47) are exactly the types of families considered in [38]. Notice that each family $\mathcal{M}'_{v_i}$ has the property that the scale of all vertices contained in $\mathcal{M}'_{v_i} - \{v_i\}$ is one larger than the scale of v_i. This property allows internal multiscale models to be realized recursively with respect to scale, *i.e.* all vectors X_v at the same scale may be designed independently of each other. A drawback to this scale-recursive approach is that a vector X_v may have to satisfy a slightly more stringent conditional independence requirement than necessary. For example, comparing the families $\mathcal{M}_6$ and $\mathcal{M}'_6$, Theorem 2.3 requires X_{13}, X_{14}, and $X_{\{3,4,5\}}$ to be conditionally independent given $X_6 = x_6$, whereas the family $\mathcal{M}'_6$ imposes a larger set of conditions.

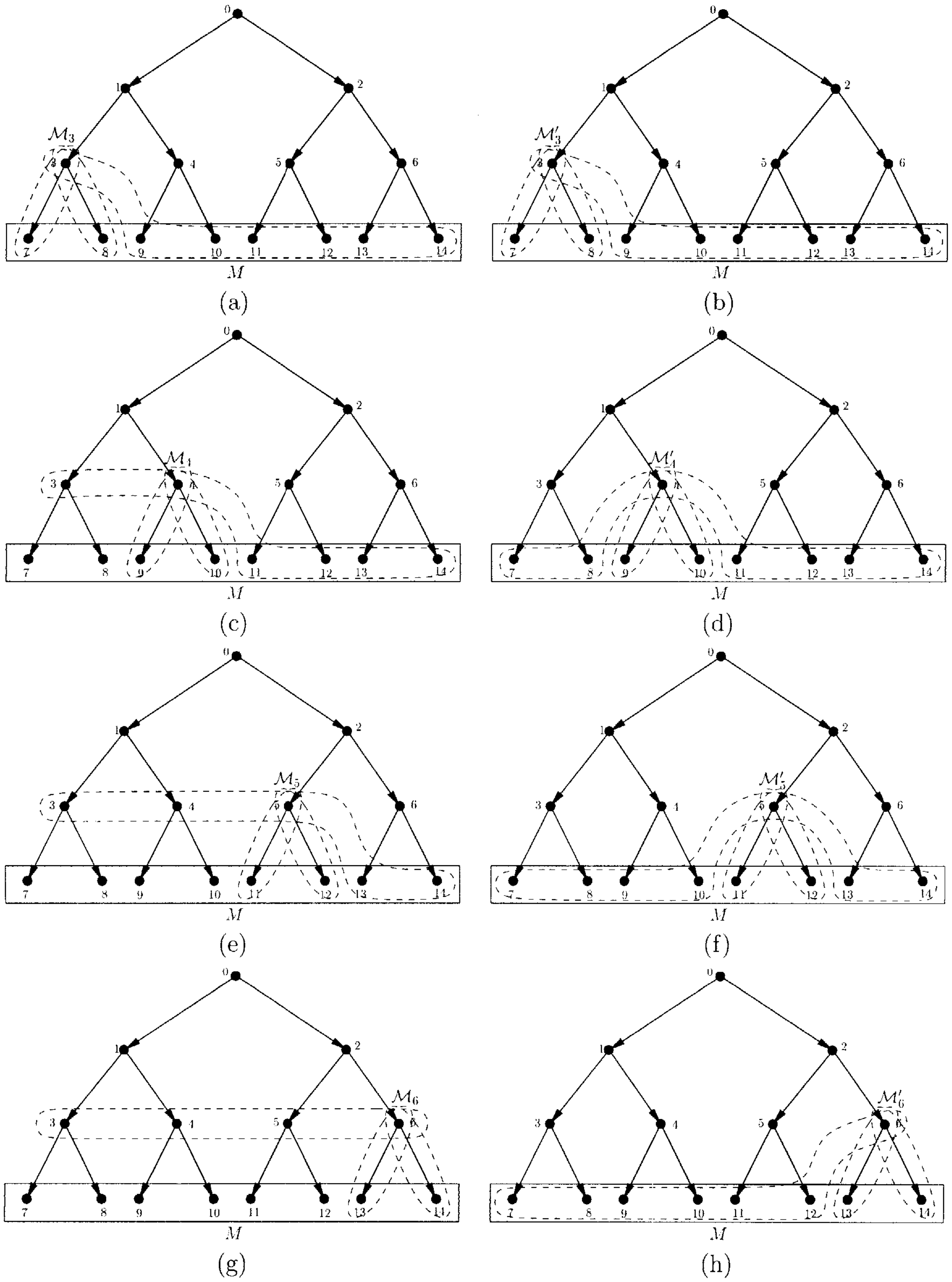


Figure 2.17. Comparison of the marginalization-invariant families (assuming the ordering $(3, 4, 5, 6, \ldots)$) and the scale-recursive families studied in [38]. The dashed contours define the sets which comprise the families $\mathcal{M}_{v_i}$ and $\mathcal{M}'_{v_i}$ provided in (2.46) and (2.47) respectively: (a) $\mathcal{M}_3$ (b) $\mathcal{M}'_3$ (c) $\mathcal{M}_4$ (d) $\mathcal{M}'_4$ (e) $\mathcal{M}_5$ (f) $\mathcal{M}'_5$ (g) $\mathcal{M}_6$ (h) $\mathcal{M}'_6$.

Chapter 3

Realizing Multiscale Models: A Graph-Theoretic Perspective

THE primary purpose of this chapter is to introduce a graph-theoretic framework for viewing the multiscale realization problem. While many of the ideas presented here apply to a broader class of modeling problems, we focus our attention on the multiscale realization problem discussed in the previous chapter, and in addition, we introduce a relaxed version of the realization problem which we call *approximate multiscale realization*. The results provided here are also used to prove several of the theorems which were stated in the previous chapter but not proven.

In Section 3.1, we formalize the multiscale realization problem for both exact and approximate models. To support our theoretical development, we provide some background material on graph theory in Section 3.2 and graphical models in Section 3.3. In Section 3.4, we return to the exact multiscale realization problem and present several alternative problem formulations along with a set of sufficient conditions for the exact realization problem, and in Section 3.5, we provide a motivating example to help guide the reader through the remaining theoretical development. The primary graph-theoretic and modeling contributions contained in this chapter are provided in Sections 3.6–3.8, and Section 3.9 uses these results to prove all of the unproven theorems from the previous chapter. Finally, Section 3.10 suggests a relaxed version of the multiscale realization problem for the purpose of realizing approximate multiscale models.

■ 3.1 General Problem Formulation

In the previous chapter, we discussed the steps involved in sequential realization of exact multiscale models. Most of the ideas presented in this chapter also relate to exact realization, but the framework we establish also applies to realizing inexact or approximate multiscale models. To motivate the need for approximations, recall that there are two goals in the exact realization problem:

(1) Given a rooted tree $\mathcal{G}_{\preceq} = (V, E)$, design a multiscale model with density $q(x_V)$ such that $q(x_M) = p^*(x_M)$ for some $M \subseteq V$.[1]

(2) Minimize the complexity of the model (and hence the computational complexity of the associated estimation algorithm).

[1] Notice that we now use the notation $p^*(x_M)$ rather than $p(x_M)$ to denote the target density to be matched. Also, instead of M, we could use $M^\sharp$, *i.e.* the marginalization constraint set associated with the state augmentation problem discussed in Section 2.7.3. For clarity, we continue to use M, with the understanding that this first step in the realization problem can be more general.

We equate the second goal with that of minimizing the dimension of each state variable to be designed in the realization problem. From the discussion in Section 2.7.1, we know that as the state dimension decreases, it becomes less likely that the first goal may be satisfied exactly, and consequently, we must be content to have $q(x_M) \approx p^*(x_M)$ in some sense. In the remainder of this section, we formalize the realization problem as well as this notion of approximation.

Recall from Section 2.3 that the tuple $(X, \mathcal{G})$ represents a process X indexed by a graph $\mathcal{G} = (V, E)$, where each X_v, $v \in V$, takes values in some space $\mathcal{X}_v$.[2] Given an indexed process $(X, \mathcal{G})$ with associated density $p(x_V)$, the notation $d(X_v)$ denotes the dimension of random vector X_v, *i.e.* $d(X_v) = n$ if and only if $X_v = (Z_1, \ldots, Z_n)^T$ with each Z_i being a scalar random variable. For each $v \in V$, we specify an integer $d_v > 0$ which represents the maximum allowable dimension of random vector X_v, *i.e.* $d(X_v) \leq d_v$ must be satisfied, and we define $d \triangleq \{d_v\}_{v \in V}$ to be the collection of all such values d_v. We say that a process $(X, \mathcal{G})$ or equivalently a density $p(x_V)$ is *consistent* with a set of dimensions d if every constraint $d(X_v) \leq d_v$, $v \in V$, is satisfied, and we define $\mathcal{P}(V, d)$ to be the set of all densities $p(x_V)$ defined on the space $\prod_{v \in V} \mathcal{X}_v$ and consistent with d,[3] *i.e.*

$$\mathcal{P}(V,d) \triangleq \left\{ p(x_V) \,\middle|\, p \text{ is a density defined on some space } \prod_{v \in V} \mathcal{X}_v,\ d(X_v) \leq d_v \ \forall v \in V \right\}. \tag{3.1}$$

Notice that a density $p \in \mathcal{P}(V, d)$ need not possess any special factorization structure and consequently such a density does not necessarily possess any conditional independencies. As discussed in the previous chapter, we are interested in the subset of densities $p \in \mathcal{P}(V, d)$ which recursively factor according to the rooted tree $\mathcal{G}_{\preceq}$, and as such, we define $\mathcal{P}_{\mathcal{G}_{\preceq}}(V, d) \subset \mathcal{P}(V, d)$ to be the set of all multiscale densities contained in $\mathcal{P}(V, d)$, *i.e.*

$$\mathcal{P}_{\mathcal{G}_{\preceq}}(V,d) \triangleq \left\{ q(x_V) \,\middle|\, q \in \mathcal{P}(V,d), q(x_V) = \prod_{v \in V} q\left(x_v | x_{\pi(v)}\right) \right\}. \tag{3.2}$$

Therefore, $\mathcal{P}_{\mathcal{G}_{\preceq}}(V, d)$ contains all possible multiscale models with the tree structure $\mathcal{G}_{\preceq}$ and with the maximum possible state dimensions allowed by d.

In order for the realization problem to be well-defined, we also require that a target density $p^*(x_M)$ be consistent with dimensions d, *i.e.* $d(X_v) \leq d_v$ for all $v \in M$. Without this requirement, the problem would be trivial in that no solution exists, and therefore, $p^*(x_M)$ is assumed to be consistent with d, even if it is not explicitly stated. Given the preceding definitions, the exact multiscale realization problem can be succinctly stated as follows:

> Exact Multiscale Realization Problem: Given a rooted tree $\mathcal{G}_{\preceq}$, a set of dimensions d, and a target density $p^*(x_M)$ consistent with d, find any density $\hat{q} \in \mathcal{P}_{\mathcal{G}_{\preceq}}(V, d)$ such that $\hat{q}(x_M) = p^*(x_M)$.

Of course, for some choices of d, there may be no density $q \in \mathcal{P}_{\mathcal{G}_{\preceq}}(V, d)$ which satisfies $q(x_M) = p^*(x_M)$, and when this occurs, we may wish to relax the exact problem formulation by finding

[2] The space $\mathcal{X}_v$ may be either a continuum or a discrete set of values. In what follows, we do not distinguish between these two cases, and for simplicity, we call $p(x_v)$ a probability density rather than a probability mass function even if $\mathcal{X}_v$ takes on a discrete set of values.

[3] We henceforth assume that the space $\prod_{v \in V} \mathcal{X}_v$ is understood from context, and we do not include this space in our notation.

the "best" approximating model according to some criterion. For the moment, suppose $D(p\|q)$ is any cost functional which provides a measure of discrimination between two densities p and q; specifically, $D(p\|q)$ discriminates between p and q in the following sense,

$$D(p\|q) \geq 0 \text{ for all densities } p \text{ and } q, \tag{3.3a}$$

$$D(p\|q) = 0 \text{ if and only if } p = q \text{ almost everywhere.} \tag{3.3b}$$

In Section 3.10.1, we discuss the *Kullback-Leibler divergence*, a cost functional satisfying (3.3) and possessing other desirable properties.

Given such a cost, the approximate multiscale realization problem may be stated as follows:

> Approximate Multiscale Realization Problem: Given a rooted tree $\mathcal{G}_{\preceq}$, a set of dimensions d, and a target density $p^*(x_M)$ consistent with d, find any density $\hat{q} \in \mathcal{P}_{\mathcal{G}_{\preceq}}(V, d)$ which minimizes the cost $D(p^*(x_M)\|\hat{q}(x_M))$, *i.e.*
>
> $$\hat{q} = \arg \min_{q \in \mathcal{P}_{\mathcal{G}_{\preceq}}(V,d)} D(p^*(x_M)\|q(x_M)). \tag{3.4}$$

Thus, instead of searching for an exact model amongst the allowable densities $\mathcal{P}_{\mathcal{G}_{\preceq}}(V, d)$, this relaxed formulation finds a density $\hat{q}$ which minimizes the cost $D(\cdot\|\cdot)$. Notice that we are interested in the complete density $\hat{q}(x_V)$ defined on all vertices V, but we only measure the discrepancy between $p^*(x_M)$ and $\hat{q}(x_M)$, *i.e.* on the desired marginal.

To develop a computationally tractable algorithm for solving the problem in (3.4), it is generally necessary to limit the set $\mathcal{P}_{\mathcal{G}_{\preceq}}(V, d)$ to some parameterized family of densities, such as all Gaussian multiscale densities. While we focus on such a parameterized family in the next chapter, in this chapter we view the realization problem from the level of abstraction in (3.4). For most of this chapter, it is assumed that there exists a $\hat{q} \in \mathcal{P}_{\mathcal{G}_{\preceq}}(V, d)$ such that $D(p^*(x_M)\|\hat{q}(x_M)) = 0$, *i.e.* an exact model, and in the final section of this chapter, the ideas developed for the exact realization problem are extended to problems where an exact model does not exist.

■ 3.2 More Graph Theory

To facilitate subsequent discussion, we now introduce additional graph-theoretic terminology. While we focus here on undirected graphs, the reader may also refer to Section 2.1 for a related discussion on directed graphs, as well as some background material assumed for our discussion here. For additional background on graph theory, there are many useful references including [16,43,71,108].

■ 3.2.1 Undirected Graphs

Recall from Section 2.1 that an undirected graph $\mathcal{G} = (V, E)$ contains only undirected edges E, *i.e.* if $(u, v) \in E$ then $(v, u) \in E$, and recall that $\mathcal{G}(U)$ is the subgraph induced by $U \subset V$. Unless otherwise indicated, we now assume that all graphs are undirected. A *complete graph* $\mathcal{G}$ on a set of vertices V is a graph which contains all possible edges (excluding self-loops). If the subgraph $\mathcal{G}(C)$ induced by $C \subseteq V$ is a complete graph, then we say that C is a *clique* of $\mathcal{G}$. If C is a clique of $\mathcal{G}$ and there does not exist a clique C' such that $C \subset C'$, we say that C is a *maximal clique* of $\mathcal{G}$. In Figure 3.1(c), for example, the sets $\{3\}$, $\{3, 7\}$, and $\{3, 7, 8\}$ all induce complete subgraphs and are therefore cliques, but the largest clique containing all of these cliques is the maximal clique $\{3, 4, 7, 8\}$.

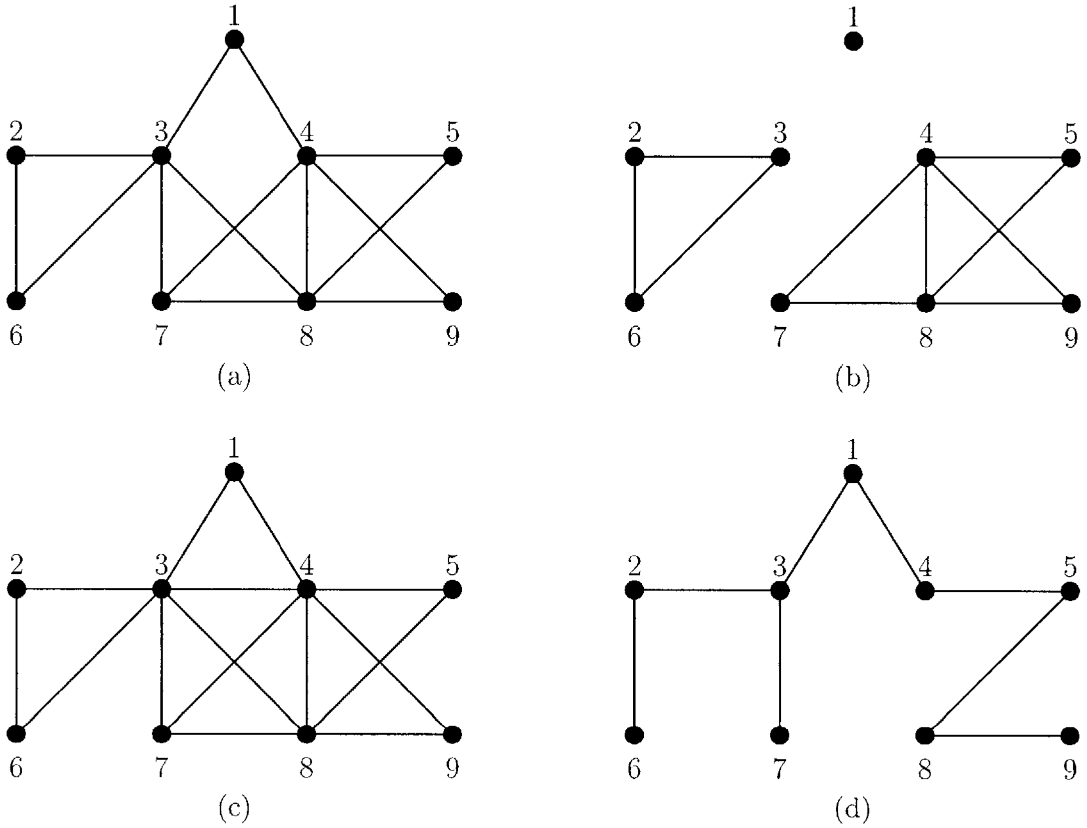


Figure 3.1. Illustrative example of several different types of graphs. (a) A graph which is not triangulated because the cycles $[1, 3, 8, 4, 1]$ and $[1, 3, 7, 4, 1]$ have no chord. (b) A graph with three connected components induced by the vertices $\{1\}$, $\{2, 3, 6\}$, and $\{4, 5, 7, 8, 9\}$. (c) A triangulated graph obtained from the non-triangulated graph in (a) by adding the edge $\{3,4\}$. (d) A spanning tree for the graph in (c).

Two vertices u and v are said to be *neighbors* in a graph $\mathcal{G} = (V, E)$ if $\{u, v\} \in E$.[4] The *neighborhood* of a vertex v is defined to be the set of all neighbors of v. We use the notation $N_{\mathcal{G}}(v)$ to represent such a neighborhood,[5] *i.e.*

$$N_{\mathcal{G}}(v) \triangleq \{u \in V |\, \mathcal{G} = (V, E),\ \{u, v\} \in E\}, \tag{3.5}$$

and for convenience, we also define $N_{\mathcal{G}}[v]$ as follows

$$N_{\mathcal{G}}[v] \triangleq N_{\mathcal{G}}(v) \cup \{v\}. \tag{3.6}$$

The preceding definitions may also be extended to a neighborhood of a set of vertices $S \subset V$,

$$N_{\mathcal{G}}[S] \triangleq \bigcup_{v \in S} N_{\mathcal{G}}[v], \tag{3.7a}$$

$$N_{\mathcal{G}}(S) \triangleq N_{\mathcal{G}}[S] - S. \tag{3.7b}$$

As an example, the graph $\mathcal{G}$ shown in Figure 3.1(d) has $N_{\mathcal{G}}(3) = \{1, 2, 7\}$, $N_{\mathcal{G}}[3] = \{1, 2, 3, 7\}$, $N_{\mathcal{G}}(\{2, 3\}) = \{1, 6, 7\}$, and $N_{\mathcal{G}}[\{2, 3\}] = \{1, 2, 3, 6, 7\}$.

The *deficiency* of vertex v in an undirected graph $\mathcal{G}$ is defined to be the set of undirected edges which must be added to $\mathcal{G}$ so that $N_{\mathcal{G}}[v]$ becomes a clique. Specifically, we define the deficiency $D_{\mathcal{G}}(v)$ as follows,

$$D_{\mathcal{G}}(v) \triangleq \{\{x, y\} |\, \mathcal{G} = (V, E),\ x, y \in N_{\mathcal{G}}(v),\ \{x, y\} \notin E\}. \tag{3.8}$$

Notice that $D_{\mathcal{G}}(v)$ is a set of sets, whereas $N_{\mathcal{G}}(v)$ is simply a set. As an example, the graph $\mathcal{G}$ shown in Figure 3.1(d) has $D_{\mathcal{G}}(3) = \{\{1, 2\}, \{1, 7\}, \{2, 7\}\}$.

If for some vertex $v \in V$, $D_{\mathcal{G}}(v) = \{\emptyset\}$, then we say v is *simplicial* in $\mathcal{G}$, and hence by definition, $N_{\mathcal{G}}[v]$ is a clique of $\mathcal{G}$. However, as the following lemma indicates, when v is simplicial, $N_{\mathcal{G}}[v]$ is not only a clique but is in fact the unique maximal clique which contains v.

Lemma 3.1 (Simplicial Vertices and Maximal Cliques).
Let $\mathcal{G} = (V, E)$ be a graph with a simplicial vertex $v \in V$. Then, $N_{\mathcal{G}}[v]$ is the unique maximal clique of $\mathcal{G}$ which contains v.

Proof. Since v is simplicial and $D_{\mathcal{G}}(v) = \{\emptyset\}$, $N_{\mathcal{G}}[v]$ is a clique in $\mathcal{G}$. If there exists another maximal clique C containing v, then there is a vertex $u \in C$ and $u \notin N_{\mathcal{G}}[v]$, such that u and v are neighbors in $\mathcal{G}$. But, then we must have $u \in N_{\mathcal{G}}[v]$ which is a contradiction. ■

Recall from Section 2.1 that a graph is connected if there exists a path between every pair of distinct vertices. If an undirected graph $\mathcal{G}$ is not connected then it contains two or more *connected components* where each connected component is a maximal connected subgraph of $\mathcal{G}$. For example, the graph shown in Figure 3.1(b) has 3 connected components induced by the vertices $\{1\}$, $\{2, 3, 6\}$, and $\{4, 5, 7, 8, 9\}$.

[4] Recall that $\{u, v\} \in E$ denotes the presence of an undirected edge in a graph.

[5] The notation $N_{\mathcal{G}}(v)$ makes it clear that the neighborhood of v is taken with respect to the graph $\mathcal{G}$. In subsequent sections, different graphs will be considered simultaneously, in which case it is important to be clear about which graph is being considered.

A subset of vertices $S \subset V$ is said to be an (a, b)-*separator* if all paths from a to b intersect S, and more generally, S is said to *separate* two sets $A \subset V$ and $B \subset V$ if it is an (a, b)-separator for every $a \in A$ and $b \in B$. In an undirected graph, this means that S separates A and B if A and B lie in different connected components when the vertices S and all incident edges are removed from the graph $\mathcal{G}$. Using the graph in Figure 3.1(a), for example, $S = \{3\}$ separates the two sets $A = \{2, 6\}$ and $B = \{1, 4, 5, 7, 8, 9\}$.

Recall that a cycle is a path which starts and ends at the same vertex. A cycle $[v_0, v_1, \ldots, v_n, v_0]$ is said to have a *chord* if $\{v_i, v_j\} \in E$ for some v_i and v_j with $1 < |i - j| < n$, and a graph is said to be *triangulated* (or *chordal*) if all cycles of length greater than three have a chord. The graph shown in Figure 3.1(a) is not triangulated because the cycles $[1, 3, 8, 4, 1]$ and $[1, 3, 7, 4, 1]$ have no chord. By adding the edge $\{3, 4\}$, as shown in Figure 3.1(c), the graph becomes triangulated.

Recall that a tree is a connected graph with no cycles. The following result indicates that there is an interesting relationship between the number of edges and the number of vertices of a tree.

Lemma 3.2 (Characterization of a Tree). *A graph $\mathcal{G} = (V, E)$ is a tree if and only if $\mathcal{G}$ is connected and $|V| = |E| + 1$.*

Proof. See [108]. ■

Given a connected undirected graph $\mathcal{G} = (V, E)$, a *spanning tree* for $\mathcal{G}$ is a tree $\mathcal{G}' = (V, E')$ such that $E' \subseteq E$, *i.e.* edges are removed from $\mathcal{G}$ to generate a tree. For example, Figure 3.1(d) shows one possible spanning tree for the graph in Figure 3.1(c).

■ 3.2.2 Junction Trees

In subsequent sections, we find great use for a special type of graph called a *junction tree*. As discussed here, junction trees are intimately related to the class of triangulated graphs, and as shown in subsequent sections, junctions trees have profound importance in the modeling problems considered in this thesis.

Given any undirected graph $\mathcal{G} = (V, E)$, the *junction graph* for $\mathcal{G}$ is defined to be the undirected graph $\mathcal{J} = (\mathcal{C}, \mathcal{E})$, whose vertex set $\mathcal{C} = \{C_1, C_2, \ldots, C_n\}$ is the collection of all maximal cliques $C_i \subset V$ of $\mathcal{G}$ and whose edge set $\mathcal{E}$ contains the undirected edge $\{C_i, C_j\} \in \mathcal{E}$ (connecting two vertices $C_i, C_j \in \mathcal{C}$) if and only if $C_i \cap C_j \neq \emptyset$.[6] As an example, consider the graph $\mathcal{G}$ in Figure 3.1(c), containing maximal cliques $\mathcal{C} \triangleq \{\{2, 3, 6\}, \{1, 3, 4\}, \{3, 4, 7, 8\}, \{4, 5, 8\}, \{4, 8, 9\}\}$. The junction graph $\mathcal{J} = (\mathcal{C}, \mathcal{E})$ for $\mathcal{G}$ is shown in Figure 3.2(a).

If an edge exists between vertices C_i and C_j in the junction graph, then a so-called *separator set* $C_i \cap C_j$ is associated with this edge, along with a *weight* equal to the size of the separator set, *i.e.* $|C_i \cap C_j|$. For the junction graph shown in Figure 3.2(a), the separator sets are included in the boxes along each edge.

Given an undirected graph $\mathcal{G}$ and corresponding junction graph $\mathcal{J}$, we now focus on spanning trees of $\mathcal{J}$ which possess a special property. Specifically, we define a *junction tree* of $\mathcal{G}$ to be any spanning tree of $\mathcal{J}$ which satisfies the following *intersection property*: for every $C_i, C_j \in \mathcal{C}$, all vertices $C_k \in \mathcal{C}$ which lie on the unique path between C_i and C_j in the spanning tree must satisfy $C_i \cap C_j \subseteq C_k$. As an example of this property, consider the graph shown in Figure 3.2(b) – a spanning tree of the junction graph shown in Figure 3.2(a). Notice that the unique path between

[6] Notice that this type of graph is more general than the graphs considered so far, since the vertices are sets and the edges represent connections between sets.

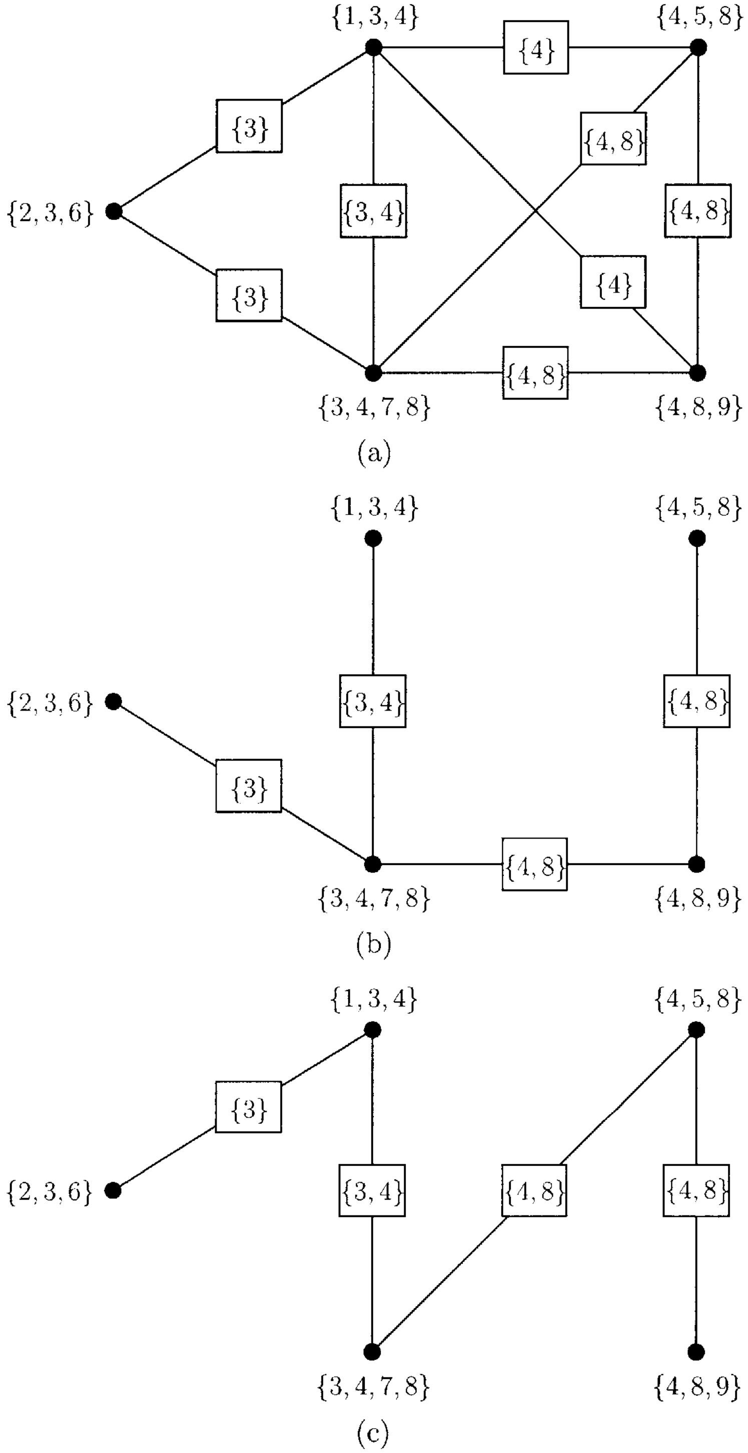


Figure 3.2. (a) The junction graph for the triangulated graph shown in Figure 3.1(c). The separator sets are shown in the boxes along each edge. (b,c) Two different spanning trees for the junction graph in (a) and consequently two possible junction trees for the triangulated graph in Figure 3.1(c).

vertices $\{1,3,4\}$ and $\{4,5,8\}$ passes through vertices $\{3,4,7,8\}$ and $\{4,8,9\}$, both of which contain $\{4\} = \{1,3,4\} \cap \{4,5,8\}$. In fact, every pair of vertices in Figure 3.2(b) satisfies this property, and consequently, the tree in Figure 3.2(b) is a junction tree for the graph in Figure 3.1(c).

While there may be many spanning trees of a junction graph, there is no guarantee that any will satisfy this intersection property. The following important theorem states that only the class of triangulated graphs will result in a spanning tree which satisfies this property.

Theorem 3.1 (Junction Trees and Triangulated Graphs).
A graph $\mathcal{G}$ is triangulated if and only if a junction tree exists for $\mathcal{G}$.

Proof. See [59]. ■

For example, since the graph shown in Figure 3.1(c) is triangulated, the junction graph in Figure 3.2(a) has a spanning tree satisfying the intersection property, as shown in Figure 3.2(b).

Another interesting property of junction trees can be seen by defining the weight of each spanning tree (for an associated junction graph) to be the sum of the weights on each edge in the tree. As the following result indicates, the set of all junction trees are those which maximize this weight.

Proposition 3.1 (Junction Trees Are Maximal Weight Spanning Trees).
Given a triangulated graph $\mathcal{G}$, a spanning tree for the junction graph of $\mathcal{G}$ is a junction tree if and only if it is a spanning tree of maximal weight.

Proof. See [54]. ■

In general, a junction graph can have a number of maximal weight spanning trees, and Proposition 3.1 therefore indicates that there may be multiple junction trees for the same triangulated graph. One might expect the separator sets associated with different junction trees to be different; however, as the following proposition indicates, this is in fact not the case.

Proposition 3.2 (Equivalence of Junction Trees).
All junction trees for a given triangulated graph have the same separator sets (also counting multiplicity).

Proof. See [56]. ■

For example, Figures 3.2(b) and (c) show two different junction trees for the triangulated graph in Figure 3.1(c), and despite the fact that the edges are different, both junction trees have the same separator sets $\{3\}$, $\{3,4\}$, $\{4,8\}$, and $\{4,8\}$. We henceforth denote the collection of separator sets by $\mathcal{S}$ (including multiplicities), so that for example $\mathcal{S} = \{\{3\},\{3,4\},\{4,8\},\{4,8\}\}$ for the junction trees shown in Figures 3.2(b) and (c).

As a consequence of the property in Proposition 3.2, we choose to define a junction tree solely in terms of its maximal cliques and its separator sets, without specifying the edges connecting the maximal cliques. Specifically, given a triangulated graph $\mathcal{G}$, we say that $\mathcal{T} = (\mathcal{C},\mathcal{S})$ is the *junction tree representation* of $\mathcal{G}$, where $\mathcal{C}$ denotes the maximal cliques of $\mathcal{G}$ and $\mathcal{S}$ denotes the separators of any junction tree for $\mathcal{G}$. Given a junction tree representation $\mathcal{T} = (\mathcal{C},\mathcal{S})$, we can always construct a junction tree by specifying a set of edges between elements of $\mathcal{C}$ such that (1) the set of separators are precisely those in $\mathcal{S}$ and (2) the graph is a tree.

3.3 Undirected Graphical Models

Graphical models provide a powerful framework for representing probability densities with particular conditional independence properties and in addition exploiting these properties in various estimation tasks. In the previous chapter, we discussed the conditional independence properties of the class of multiscale models – a special type of graphical model defined on a directed graph. In this section, we focus on graphical models defined on undirected graphs, and we discuss their associated conditional independence properties. We also introduce an important factorization for triangulated graphs which uses the junction tree representation, and finally, we show how multiscale models can also be interpreted and represented as undirected graphical models. A more detailed treatment of graphical models may be found in a variety of sources [41,55,58,59,71].

3.3.1 Undirected Graphical Models and Their Conditional Independence Properties

Suppose an indexed process $(X, \mathcal{G})$ is given. By itself, such a process is rather uninteresting; however, if X possesses a set of conditional independencies or if the associated density $p(x)$ has a special factorization structure then things become more interesting. For example, as discussed in Sections 2.3 and 2.4, there exists an important relationship between factorization and conditional independence properties for the class of multiscale models. As we now discuss, a similar relationship exists for processes mapped to undirected graphs.

Suppose $\mathcal{G}$ is an undirected graph with a set of maximal cliques $\mathcal{C}$, and let $(X, \mathcal{G})$ be a process indexed by $\mathcal{G}$ with an associated probability density p. Consider the following important factorized form for a density.

Definition 3.1 (Factorization According to an Undirected Graph).
A density $p(x)$ is said to *factor according to* $\mathcal{G}$ if there exist non-negative functions $\psi_C(x_C)$, $C \in \mathcal{C}$, such that

$$p(x) = \prod_{C \in \mathcal{C}} \psi_C(x_C). \tag{3.9}$$

◀

The functions $\psi_C(x_C)$ are often called *compatibility functions*. Notice that each function $\psi_C(x_C)$ depends only on the variables indexed by the maximal clique C, and consequently, the density p is a product of "localized" functions, where the degree of localization is determined by the graph $\mathcal{G}$.

Given a density p which factors according to (3.9), it must be true that random vector X displays some set of independencies; otherwise, no factorization would be possible. The following definition provides a very compact and convenient statement of the conditional independencies possessed by X, a fact which is later proven in Theorem 3.2.

Definition 3.2 (Markov Properties and Graph Separation).
An indexed process $(X, \mathcal{G})$ is said to be *Markov* with respect to the undirected graph $\mathcal{G}$ if X_A and X_B are conditionally independent given X_S whenever A and B are separated by S in $\mathcal{G}$. ◀

Therefore, if a process is Markov with respect to a graph $\mathcal{G}$, all of the conditional independencies can be read directly from $\mathcal{G}$ by examining the separation properties of the graph. This is an important aspect of graphical models, since it allows graph-theoretic results to be brought to bear on the modeling problem. As an example, consider the graphical model shown in Figure 3.3(a), where the underlying graph $\mathcal{G}$ corresponds to a simple cycle. If a process X is Markov with respect to $\mathcal{G}$, then X_1 and X_4 are conditionally independent given X_2 and X_3, due to the fact that $S = \{2,3\}$ separates $A = \{1\}$ and $B = \{4\}$ in $\mathcal{G}$.

The following important theorem indicates that a density of the form (3.9) will have precisely the conditional independencies given in Definition 3.2. One might wonder if the converse is also true. The answer is yes but with the caveat that the density $p(x)$ is *strictly positive*, *i.e.* if an outcome $x = (x_1, \dots, x_n)$ satisfies $p(x_i) > 0$ for $i = 1, \dots, n$, then we must have $p(x) > 0$ [10].

Theorem 3.2 (Hammersley-Clifford [45]).
Let $\mathcal{G}$ be an undirected graph with maximal cliques $\mathcal{C}$, and let $(X, \mathcal{G})$ be an indexed process with density $p(x)$. If $p(x)$ factors according to (3.9) then $(X, \mathcal{G})$ is Markov with respect to $\mathcal{G}$. Conversely, if $p(x)$ is strictly positive and $(X, \mathcal{G})$ is Markov with respect to $\mathcal{G}$, then $p(x)$ factors according to (3.9).

Proof. See [10, 44]. ■

Therefore, as long as the density $p(x)$ is strictly positive, the notions of factorization and Markovianity specified in Definitions 3.1 and 3.2 are equivalent.

Notice that there exists an important difference between undirected graphical models and directed acyclic graphical models (such as multiscale models). Namely, the factorization given in (3.9) is stated in terms of abstract functions $\psi_C(x_C)$ and not probability densities as is the case for directed models. As discussed in the next section, though, there are undirected graphical models for which the compatibility functions may be written in terms of probability densities.

■ 3.3.2 An Important Factorization

In this section, we discuss an important factorization of the form (3.9), where each $\psi_C(x_C)$ is expressed solely as a function of $p(x_C)$ and marginals of $p(x_C)$. It turns out that such a factorization does not generally exist when the underlying graph $\mathcal{G}$ is not triangulated, while such a factorization always exists when $\mathcal{G}$ is triangulated. To develop some intuition for this fact, consider the following example.

Example 3.1 (Factorization and the Single Cycle).
Consider the graphical model shown in Figure 3.3(a) where the graph $\mathcal{G}$ corresponds to a 4-cycle and the associated density p factors according to

$$p(x_1, x_2, x_3, x_4) = \psi_{\{1,2\}}(x_1, x_2)\psi_{\{1,3\}}(x_1, x_3)\psi_{\{2,4\}}(x_2, x_4)\psi_{\{3,4\}}(x_3, x_4), \tag{3.10}$$

for some choice of compatibility functions. Our goal is to specify each of the compatibility functions $\psi_C(x_C)$ in (3.10) directly in terms of $p(x_C)$ and its marginals.

Consider expanding p using the chain rule for probabilities,

$$p(x_1, x_2, x_3, x_4) = p(x_1, x_2, x_3)p(x_4|x_1, x_2, x_3). \tag{3.11}$$

This expression may be simplified by applying the Markov properties implied by $\mathcal{G}$,

$$\begin{aligned} p(x_1, x_2, x_3, x_4) &= p(x_1, x_2, x_3)p(x_4|x_2, x_3) \\ &= p(x_1, x_2, x_3)\left(\frac{p(x_2, x_3, x_4)}{p(x_2, x_3)}\right). \end{aligned} \tag{3.12}$$

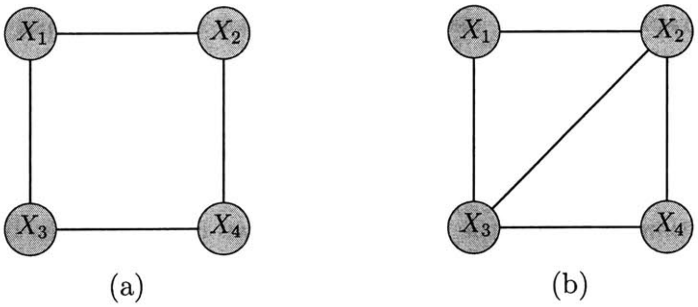


Figure 3.3. (a) A graphical model defined on a non-triangulated graph. (b) A graphical model defined on a triangulated graph.

The terms $p(x_1, x_2, x_3)$ and $p(x_2, x_3, x_4)$ cannot be simplified further because the graphical model shown in Figure 3.3(a) does not imply any additional independencies, and since these two densities are functions of three variables, (3.12) does not fit the desired form in (3.10). Of course, the chain rule in (3.11) may be applied in multiple ways; however, every factorization ultimately requires a marginal of the form $p(x_i, x_j, x_k)$. The reason that the desired factorization (3.10) is not achievable is due to the fact that $\mathcal{G}$ is not triangulated.

Consider now the graphical model shown in Figure 3.3(b), with a density which factors according to

$$p(x_1, x_2, x_3, x_4) = \psi_{\{1,2,3\}}(x_1, x_2, x_3)\psi_{\{2,3,4\}}(x_2, x_3, x_4). \tag{3.13}$$

Using the decomposition in (3.11) and applying the Markov properties of this new model, we again obtain the decomposition in (3.12). In this case, (3.12) fits the desired form in (3.13), due to fact that the graphical model in Figure 3.3(b) is indexed by a triangulated graph. Since the marginal $p(x_2, x_3)$ in (3.12) may be shared between $\psi_{\{1,2,3\}}(x_1, x_2, x_3)$ and $\psi_{\{2,3,4\}}(x_2, x_3, x_4)$ in multiple ways, there is no unique choice for the compatibility functions. For our purposes, this non-uniqueness is inconsequential because we only care about the form of the factorization in (3.12) and not about a particular choice for the compatibility functions. ◀

The factorization in (3.12) can be generalized to hold for any triangulated graph [59]. Specifically, let $\mathcal{G}$ be a triangulated graph with the junction tree representation $\mathcal{T} = (\mathcal{C}, \mathcal{S})$, and let p be a density which factors according to $\mathcal{G}$. Then, p may be expressed as follows,

$$p(x) = \prod_{C\in\mathcal{C}} \psi_C(x_C) = \frac{\prod_{C\in\mathcal{C}} p(x_C)}{\prod_{S\in\mathcal{S}} p(x_S)}. \tag{3.14}$$

Since the junction tree representation $\mathcal{T} = (\mathcal{C}, \mathcal{S})$ is unique for a given triangulated graph, the preceding decomposition is well-defined, *i.e.* it does not depend on a particular choice of junction tree. In addition, since $\mathcal{G}$ is triangulated, Theorem 3.2 holds even when the density $p(x)$ is not strictly positive [99]. Thus, if X is Markov with respect to a triangulated graph $\mathcal{G}$, the corresponding density $p(x)$ must factor according to (3.14).

Consider now a slightly different perspective on the factorization in (3.14). Suppose an indexed process $(X, \mathcal{G})$ with density $p(x)$ is given, and further suppose that $p(x)$ does not factor according to the triangulated graph $\mathcal{G}$. Then, using (3.14), we can define a density $p_{\mathcal{G}}$ which does factor according to $\mathcal{G}$ as follows,

$$p_{\mathcal{G}}(x) \triangleq \frac{\prod_{C \in \mathcal{C}} p(x_C)}{\prod_{S \in \mathcal{S}} p(x_S)}. \tag{3.15}$$

We call $p_{\mathcal{G}}$ the *projection* of p onto the graph $\mathcal{G}$,[7] and we take (3.15) to be the definition of this projection. If $\mathcal{G}^{\downarrow} = (U, F)$ is any subgraph of $\mathcal{G} = (V, E)$ and $p(x_V)$ is a density indexed by V, then the notation $p_{\mathcal{G}^{\downarrow}}(x_U)$ indicates the projection of the marginal $p(x_U)$ onto the subgraph $\mathcal{G}^{\downarrow}$. For notational convenience, we often write $p_{\mathcal{G}^{\downarrow}}(x)$ instead of $p_{\mathcal{G}^{\downarrow}}(x_U)$, with the understanding that the argument x is indexed by the vertices of the graph $\mathcal{G}^{\downarrow}$.

An interesting consequence of (3.15) is that $p_{\mathcal{G}}$ satisfies a set of marginal constraints. Namely, for every maximal clique $C \in \mathcal{C}$, the constraint $p_{\mathcal{G}}(x_C) = p(x_C)$ is satisfied. This follows from the fact that $p_{\mathcal{G}}$ factors according to $\mathcal{G}$, in which case

$$p_{\mathcal{G}}(x) = \frac{\prod_{C \in \mathcal{C}} p_{\mathcal{G}}(x_C)}{\prod_{S \in \mathcal{S}} p_{\mathcal{G}}(x_S)} = \frac{\prod_{C \in \mathcal{C}} p(x_C)}{\prod_{S \in \mathcal{S}} p(x_S)}.$$

Hence, we can view $p_{\mathcal{G}}$ as the density which factors according to $\mathcal{G}$ while maintaining the correct marginal densities $p(x_C)$ on the maximal cliques. We use this idea extensively in subsequent sections.

■ 3.3.3 A Special Case: Multiscale Models

Multiscale Models and Undirected Graphs

In Chapter 2, multiscale models were defined in terms of a recursive factorization with respect to a rooted tree. In this section, we show how multiscale models may also be defined in terms of a factorization with respect to an undirected graph. To view this equivalence, we compare the multiscale factorization to the factorization given in (3.14). Specifically, let $\mathcal{G}_{\preceq} = (V, E)$ be a rooted tree, and let $q(x)$ be a multiscale density with the usual factorization,

$$q(x) = \prod_{v \in V} q(x_v | x_{\pi(v)}).$$

Letting v_0 correspond to the root vertex, $q(x)$ can be rewritten as follows,

$$q(x) = q(x_{v_0}) \prod_{v \in V - \{v_0\}} \frac{q(x_v, x_{\pi(v)})}{q(x_{\pi(v)})}. \tag{3.16}$$

Notice that (3.16) resembles the form of (3.14) with $\{v, \pi(v)\}$ corresponding to a maximal clique and $\{\pi(v)\}$ corresponding to a separator set.

To solidify this relationship, let the graph $\mathcal{G}_{\preceq}^{\sim}$ represent the *undirected version* of $\mathcal{G}_{\preceq}$, *i.e.* every directed edge in $\mathcal{G}_{\preceq}$ is replaced by an undirected edge. Consequently, $\mathcal{G}_{\preceq}^{\sim}$ is an undirected graph which is a tree. For illustrative purposes, a rooted tree $\mathcal{G}_{\preceq}$ and its undirected version $\mathcal{G}_{\preceq}^{\sim}$ are

[7] As we later discuss, this terminology is appropriate from a geometric perspective.

shown in Figures 3.4(a) and (b) respectively. It can be shown that the junction tree representation $\mathcal{T} = (\mathcal{C}, \mathcal{S})$ for $\mathcal{G}^{\sim}_{\preceq}$ is given by[8]

$$\mathcal{C} = \{\{v, \pi(v)\}\}_{v \in V - \{v_0\}} \tag{3.17a}$$

$$\mathcal{S} = \{\{\pi(v)\}\}_{v \in V - \{v_0\}} - \{\{v_0\}\}. \tag{3.17b}$$

Notice that one of the root vertices must be removed from the separator sets $\mathcal{S}$ due to an overcounting. As an example, Figure 3.4(c) shows a junction tree for the graph in Figure 3.4(b).

Given the junction tree representation in (3.17), any density q which factors according to $\mathcal{G}^{\sim}_{\preceq}$ may be written as follows,

$$q(x) = \frac{\prod_{v \in V - \{v_0\}} q(x_v, x_{\pi(v)})}{\frac{1}{q(x_{v_0})} \prod_{v \in V - \{v_0\}} q(x_{\pi(v)})}. \tag{3.18}$$

Since (3.16) and (3.18) are equivalent factorizations, we know that a multiscale density $q(x)$ defined on a rooted tree $\mathcal{G}_{\preceq}$ is equivalent to a density $q(x)$ which factors according $\mathcal{G}^{\sim}_{\preceq}$. This equivalence also implies that the conditional independencies exhibited by multiscale models must be identical to the conditional independencies given in Definition 3.2. This is of course true because the global Markov property of multiscale models was stated precisely in terms of graph separation, *i.e.* the random vectors in the subtrees separated by a vertex v are conditionally independent given X_v.

Hence, multiscale models can be characterized either in terms of a directed factorization or an undirected factorization. Recall that the benefit of the directed factorization is two-fold: (1) a directed factorization leads to a simple "parametrization" in terms of conditional probabilities mapped to edges of the graph; (2) the rooted tree possesses the notion of scale which induces a partial ordering on the graph. The reason to now consider multiscale models in terms of their undirected factorization is that this factorization allows more flexibility and fits more easily within the framework established in this chapter.

Multiscale Models and Projections

Given any density $p(x)$, not necessarily a multiscale density, we can always form a multiscale density $q(x)$ by projecting $p(x)$ onto a graph which is a tree. Specifically, given a rooted tree $\mathcal{G}_{\preceq}$ and corresponding undirected version $\mathcal{G}^{\sim}_{\preceq}$, a multiscale density $q(x)$ may be formed by projecting any density $p(x)$ onto the graph $\mathcal{G}^{\sim}_{\preceq}$ according to (3.15). Since $\mathcal{G}^{\sim}_{\preceq}$ has the junction tree representation in (3.17), $q(x)$ is given by,

$$q(x) \triangleq p_{\mathcal{G}^{\sim}_{\preceq}}(x) = p(x_{v_0}) \prod_{v \in V - \{v_0\}} \frac{p(x_v, x_{\pi(v)})}{p(x_{\pi(v)})} = \prod_{v \in V} p(x_v | x_{\pi(v)}). \tag{3.19}$$

Notice that (3.19) is the same as (2.25); this equivalence provides additional intuition for the realization approach suggested in the previous chapter.

When the underlying rooted tree $\mathcal{G}_{\preceq}$ is clear from context, we often use the notation $p^T(x)$ to denote the multiscale density formed by projecting $p(x)$ onto the tree $\mathcal{G}^{\sim}_{\preceq}$, *i.e.* $p^T(x) \triangleq p_{\mathcal{G}^{\sim}_{\preceq}}(x)$. This projection operation also provides a mapping $p \longrightarrow p^T$ from $\mathcal{P}(V, d)$ to $\mathcal{P}_{\mathcal{G}_{\preceq}}(V, d)$, where

[8] For convenience, we retain the labeling implied by the rooted tree, *i.e.* even though $\mathcal{G}^{\sim}_{\preceq}$ is undirected, we still use the notation $\pi(v)$ as if $\mathcal{G}^{\sim}_{\preceq}$ were the rooted tree $\mathcal{G}_{\preceq}$.

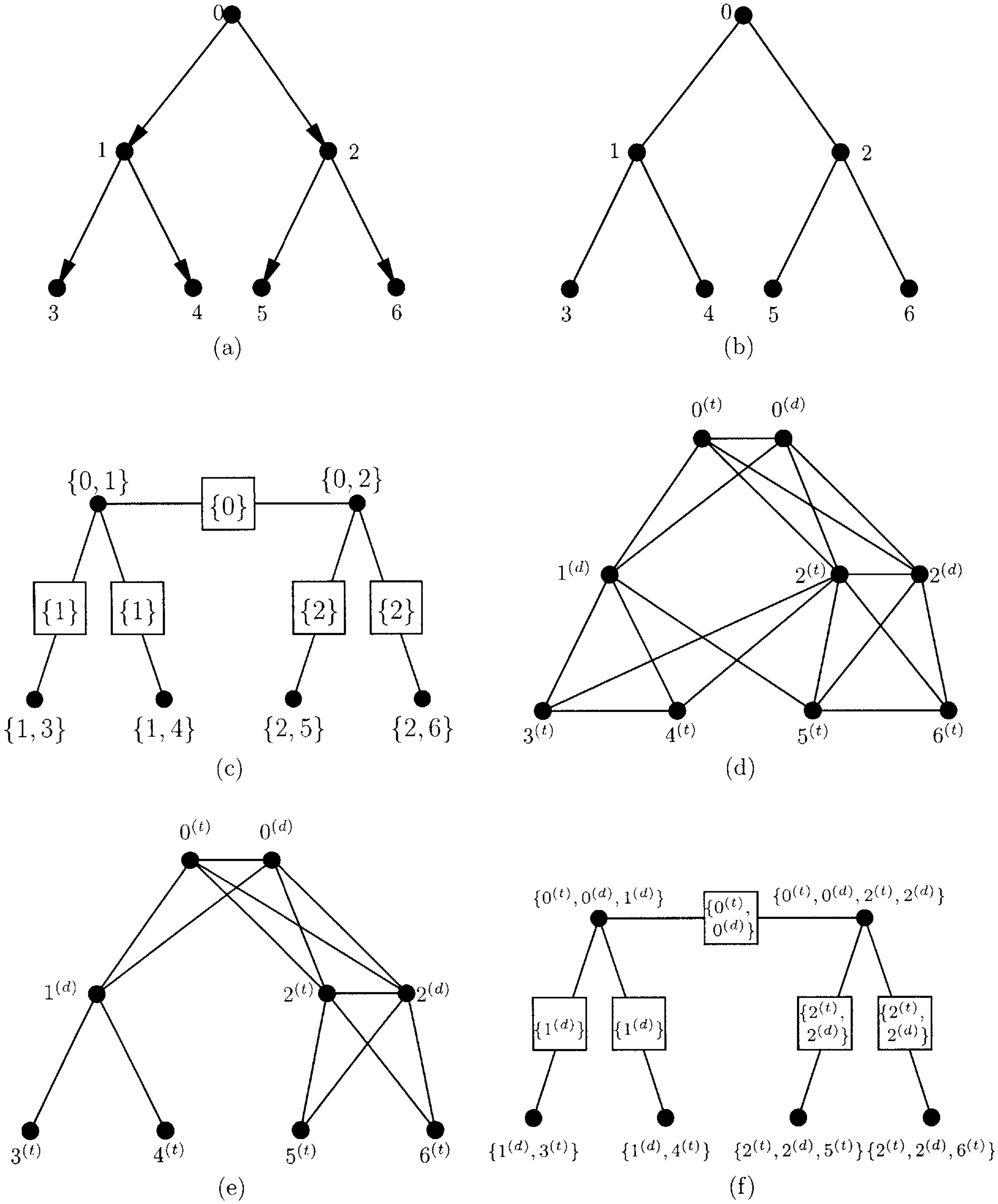


Figure 3.4. (a) A rooted tree $\mathcal{G}_{\preceq}$. (b) The undirected version of the rooted tree in (a). (c) A junction tree for the graph in (b). (d) An augmented graph $\mathcal{G}^{\sharp}$ for the rooted tree in (a), assuming $M = \{0, 2, 3, 4, 5, 6\}$. (e) The augmented graph $\mathcal{G}^{\sharp}_{\preceq}$ for the rooted tree in (a), assuming $M = \{0, 2, 3, 4, 5, 6\}$. (f) A junction tree for the graph in (e).

$\mathcal{P}(V,d)$ and $\mathcal{P}_{\mathcal{G}_{\preceq}}(V,d)$ are the sets of densities introduced in Section 3.1. In other words, given any $p \in \mathcal{P}(V,d)$, the density p^T (where the projection is with respect to $\mathcal{G}_{\preceq}^{\sim}$) satisfies $p^T \in \mathcal{P}_{\mathcal{G}_{\preceq}}(V,d)$. We focus on this mapping in more detail in Section 3.4.

Multiscale Models and Augmented Graphs

In the previous chapter, we introduced a more general type of multiscale model where each state variable X_v is allowed to have both a target vector $X_{v^{(t)}}$ as well as a design vector $X_{v^{(d)}}$. From a graphical modeling perspective, splitting X_v into two sub-vectors $X_{v^{(d)}}$ and $X_{v^{(t)}}$ provides additional degrees of freedom in the realization problem and at the same time allows us to consider models with more complicated dependencies. To incorporate this indexing scheme into our graph-theoretic framework, it is convenient to introduce a special type of graph which we call an *augmented graph* and often denote by $\mathcal{G}^{\sharp}$. Augmented graphs are the same as the undirected graphs considered so far but with the modification that both types of vertices $v^{(d)}$ and $v^{(t)}$ are present.

To define an augmented graph $\mathcal{G}^{\sharp}$, a rooted tree $\mathcal{G}_{\preceq} = (V,E)$ and a marginalization constraint set M must be specified,[9] and therefore, every augmented graph $\mathcal{G}^{\sharp}$ is tied to an underlying tree $\mathcal{G}_{\preceq}$ and some $M \subseteq V$.[10] Once $\mathcal{G}_{\preceq}$ and M are specified, an augmented graph $\mathcal{G}^{\sharp}$ is defined to be any undirected graph on a set of vertices $V^{\sharp} \subset \{v^{(d)}, v^{(t)}\}_{v \in V}$ which satisfies the following constraints:

(1) $v^{(t)} \in V^{\sharp}$ if and only if $v \in M$;

(2) $v^{(d)} \in V^{\sharp}$ if and only if v is a non-leaf vertex in $\mathcal{G}_{\preceq}$.

Figure 3.4(d) shows one example of an augmented graph for the rooted tree $\mathcal{G}_{\preceq}$ in Figure 3.4(a) and $M = \{0,2,3,4,5,6\}$.

We impose the preceding constraints on an augmented graph in order to avoid superfluous vertices that index meaningless vectors in the corresponding graphical model. Specifically, recall that by definition no target vector $X_{v^{(t)}}$ exists for a vertex $v \notin M$, and consequently, when $v \notin M$, vertex $v^{(t)}$ is not included in the augmented graph. Similarly, a design vector $X_{v^{(d)}}$ located at a leaf vertex of a multiscale model serves no purpose in the realization problem, and therefore, the second constraint indicates that $v^{(d)}$ is not included in the augmented graph if v is a leaf vertex of $\mathcal{G}_{\preceq}$.

We now focus on a specific augmented graph denoted by $\mathcal{G}_{\preceq}^{\sharp}$. For a given rooted tree $\mathcal{G}_{\preceq}$, the graph $\mathcal{G}_{\preceq}^{\sharp} = (V^{\sharp}, E^{\sharp})$ is defined to be the augmented graph with edge set $E^{\sharp}$, where $\{x,y\} \in E^{\sharp}$ if and only if one of the following is satisfied:

(1) $x = v^{(t)}$ and $y = v^{(d)}$;

(2) $x = u^{(t)}$ or $x = u^{(d)}$, $y = v^{(t)}$ or $y = v^{(d)}$, and $u \in \pi(v)$.

For example, Figure 3.4(e) shows the graph $\mathcal{G}_{\preceq}^{\sharp}$ for the rooted tree $\mathcal{G}_{\preceq}$ in Figure 3.4(a) and $M = \{0,2,3,4,5,6\}$.

The augmented graph $\mathcal{G}_{\preceq}^{\sharp}$ satisfies an important property: a process X which is Markov with respect to $\mathcal{G}_{\preceq}^{\sim}$ is also Markov with respect to the graph $\mathcal{G}_{\preceq}^{\sharp}$. To see this, suppose we are given

[9] We assume here that M contains at least all of the leaf vertices of $\mathcal{G}_{\preceq}$.

[10] The notation $\mathcal{G}^{\sharp}$ does not convey this dependence on $\mathcal{G}_{\preceq}$ and M, but the particular choice of $\mathcal{G}_{\preceq}$ and M will always be clear.

a density $q(x_V)$ indexed by a set of vertices V, and consider splitting each variable x_v in $q(x_V)$ into separate target and design variables $x_{v^{(t)}}$ and $x_{v^{(d)}}$. This means that we can also index the density $q(x_V)$ in terms of the expanded vertex set $V^{\sharp}$, which we denote by $q(x_{V^{\sharp}})$. Consider now a multiscale density $q(x_V)$ which factors according to the graph $\mathcal{G}_{\preceq}^{\sim} = (V, E)$. The structure of the graph $\mathcal{G}_{\preceq}^{\sharp} = (V^{\sharp}, E^{\sharp})$ is special in the sense that the re-indexed density $q(x_{V^{\sharp}})$ also factors according to $\mathcal{G}_{\preceq}^{\sharp}$, and consequently, the graph $\mathcal{G}_{\preceq}^{\sharp}$ implies the same conditional independencies as $\mathcal{G}_{\preceq}^{\sim}$.

As an illustration, consider the augmented graph $\mathcal{G}_{\preceq}^{\sharp}$ and corresponding junction tree shown in Figures 3.4(e) and (f) respectively. Notably, the junction tree in Figure 3.4(f) is structurally identical to the junction tree shown in Figure 3.4(c); the only difference is that each vertex v has been replaced by its appropriate label $v^{(d)}$ and/or $v^{(t)}$. More generally, the junction trees for the graphs $\mathcal{G}_{\preceq}^{\sim}$ and $\mathcal{G}_{\preceq}^{\sharp}$ are always structurally identical (by the construction of $\mathcal{G}_{\preceq}^{\sharp}$), and consequently, a density q which factors according to $\mathcal{G}_{\preceq}^{\sharp}$ also factors according to $\mathcal{G}_{\preceq}^{\sim}$ and vice-versa.

Simply stated, augmented graphs are undirected graphs defined on an expanded vertex set. They are useful for our purposes because they allow the two types of vectors $X_{v^{(t)}}$ and $X_{v^{(d)}}$ to be indexed in a simple manner. As we later show, the graph-theoretic and graphical modeling ideas presented here directly translate to augmented graphs, and therefore, we are able to state several important results about the state augmentation problem without much additional work.

■ 3.4 Alternative Problem Formulations For Exact Realization

In Section 3.1, we introduced an abstract statement of the exact multiscale realization problem, which we now term *exact multiscale realization problem* $\mathcal{Q}$ and include below for reference.

> Exact Multiscale Realization Problem $\mathcal{Q}$: Find any density $\hat{q} \in \mathcal{P}_{\mathcal{G}_{\preceq}}(V, d)$ such that $\hat{q}(x_M) = p^*(x_M)$.

In this section, we introduce a number of alternative problems, each of which is equivalent to the preceding problem in the sense that a surjective mapping exists from the solution set of each alternative problem to the solution set of problem $\mathcal{Q}$.[11] Hence, by solving one of these alternative problems, we are able to identify a solution to problem $\mathcal{Q}$ via this surjective mapping. We also show that there exists an interesting relationship between the solution sets of some of the proposed alternative problems, and we use this relationship to suggest a number of different sufficient conditions for the exact realization problem.

■ 3.4.1 Two Alternative Problem Formulations

As an alternative to realization problem $\mathcal{Q}$, consider the problem of searching amongst a larger set of densities – namely searching in the full set of densities $\mathcal{P}(V, d)$ rather than the set of multiscale densities $\mathcal{P}_{\mathcal{G}_{\preceq}}(V, d)$. The goal is to find a density $\hat{p} \in \mathcal{P}(V, d)$ such that the projection $\hat{p}^T$ onto the tree $\mathcal{G}_{\preceq}^{\sim}$ satisfies $\hat{p}^T(x_M) = p^*(x_M)$.[12] We call this procedure *alternative problem* $\mathcal{P}$.

[11] Of course, the exact realization problem $\mathcal{Q}$ might have no solution, in which case all of the proposed alternative problems will also have no solution. In this situation, we must consider the approximate realization problem introduced in Section 3.1 and discussed in more detail in Section 3.10.

[12] Notice that $\hat{p}^T(x_M)$ is a shorthand for the marginal of $\hat{p}^T(x)$ along the variables x_M, *i.e.* $\hat{p}^T(x_M) = \hat{q}(x_M)$ where $\hat{q}(x) \triangleq \hat{p}^T(x)$.

> Alternative Problem $\mathcal{P}$: Find any density $\hat{p} \in \mathcal{P}(V, d)$ such that $\hat{p}^T(x_M) = p^*(x_M)$.

Since problem $\mathcal{P}$ searches over a larger space of densities, it is more computationally demanding than problem $\mathcal{Q}$. At the same time, this larger problem is theoretically interesting due to the fact that the mapping $p \longrightarrow p^T$ is a surjection from the solution set of problem $\mathcal{P}$ onto the solution set of problem $\mathcal{Q}$. This implies that problem $\mathcal{P}$ may be considered without loss of generality, since all solutions of problem $\mathcal{Q}$ can be identified from solutions of problem $\mathcal{P}$. Because of this fact, we say that problems $\mathcal{P}$ and $\mathcal{Q}$ are *compatible.*

Recall that the projection p^T was previously introduced in Section 3.3.3. For a given rooted tree $\mathcal{G}_{\preceq}$, the mapping $p \longrightarrow p^T$ is a surjection from the space of densities $\mathcal{P}(V, d)$ onto $\mathcal{P}_{\mathcal{G}_{\preceq}}(V, d)$, since every $q \in \mathcal{P}_{\mathcal{G}_{\preceq}}(V, d)$ satisfies $q \in \mathcal{P}(V, d)$ and $q = q^T$. At the same time, this mapping is by definition a surjection from the solution set of problem $\mathcal{P}$ onto the solution set of $\mathcal{Q}$. Since it is surjective, though, there may be infinitely many solutions $\hat{p}$ each of which map to the same density $\hat{p}^T$, and consequently, the solution set for problem $\mathcal{P}$ may be much larger than the solution set for problem $\mathcal{Q}$.

For our purposes, problem $\mathcal{P}$ searches over a larger space of densities than necessary, and as such, in the remainder of this section and in the next section we further limit the space of densities over which we search. Define $\mathcal{P}^M(V, d) \subset \mathcal{P}(V, d)$ to be the set of all densities $p \in \mathcal{P}(V, d)$ with the marginal $p(x_M) = p^*(x_M)$, *i.e.*

$$\mathcal{P}^M(V, d) \triangleq \{p(x_V) | p \in \mathcal{P}(V, d), p(x_M) = p^*(x_M)\}. \tag{3.20}$$

The following procedure, which we call *alternative problem* $\mathcal{P}^M$, is identical to problem $\mathcal{P}$ except we search over the set $\mathcal{P}^M(V, d)$ instead of $\mathcal{P}(V, d)$.

> Alternative Problem $\mathcal{P}^M$: Find any density $\hat{p} \in \mathcal{P}^M(V, d)$ such that $\hat{p}^T(x_M) = p^*(x_M)$.

Notice that $p \longrightarrow p^T$ is also a surjection from the solution set of $\mathcal{P}^M$ onto the solution set of $\mathcal{Q}$, and hence, problems $\mathcal{P}^M$ and $\mathcal{Q}$ are compatible. This relies on the fact that every solution $\hat{q}$ of $\mathcal{Q}$ is also a solution to $\mathcal{P}^M$ (since $\hat{q}(x_M) = p^*(x_M)$ and therefore $\hat{q} \in \mathcal{P}^M(V, d)$).

One immediate benefit of considering problem $\mathcal{P}^M$ instead of $\mathcal{P}$ is the fact that we can state a simple sufficient condition for a density $\hat{p}$ to be a solution to problem $\mathcal{P}^M$. Specifically, if $\hat{p} \in \mathcal{P}^M(V, d)$ satisfies $\hat{p} = \hat{p}^T$, then $\hat{p}$ is a solution to problem $\mathcal{P}^M$. This follows trivially from the fact that (1) $\hat{p}(x_M) = p^*(x_M)$ since $\hat{p} \in \mathcal{P}^M(V, d)$ and (2) $\hat{p}(x_M) = \hat{p}^T(x_M) = p^*(x_M)$ since $\hat{p} = \hat{p}^T$. Notice that solutions to problem $\mathcal{P}$ cannot be identified in such a manner, *i.e.* even if $p = p^T$, there is no constraint on the marginal $p(x_M)$.

While the preceding is a sufficient condition for solutions to problem $\mathcal{P}^M$, it is not necessary – there may exist an infinite number of solutions $\hat{p} \in \mathcal{P}^M(V, d)$ which project to the same density $\hat{p}^T$ but do not satisfy $\hat{p} = \hat{p}^T$. In fact, this sufficient condition simply characterizes all solutions to problem $\mathcal{Q}$, and for this reason, it is not immediately obvious from the preceding discussion why a more general problem formulation should be considered. The utility lies in the fact that a less-stringent set of sufficient conditions may be obtained by considering a more structured problem than $\mathcal{P}^M$.

To develop some intuition and at the same time tie the discussion here to the discussion in Chapter 2, recall that the condition $\hat{p} = \hat{p}^T$ is equivalent to requiring random vector X to satisfy the global Markov property with respect to density $\hat{p}$. As stated in Theorem 2.3 (but not yet proven), when we only care about matching a marginal, *i.e.* $\hat{p}(x_M) = \hat{p}^T(x_M)$, there exists a smaller set of sufficient conditions than those required by the global Markov property (or equivalently the reduced-order global Markov property). This suggests that the conditions $\hat{p} \in \mathcal{P}^M(V, d)$ and $\hat{p} = \hat{p}^T$ are too stringent because they place constraints on the entire density $\hat{p}$. By considering alternatives to problem $\mathcal{P}^M$, as discussed in the next section, we subsequently show in Section 3.4.3 that less-stringent sufficient conditions exist.

■ 3.4.2 More Possibilities

The problem formulations considered in this section differ from those considered in the previous section in that they possess additional conditional independence structure. Each problem has the same form as problem $\mathcal{P}^M$, except the search is performed over a more limited space of densities with a particular set of independencies. The set of independencies and the associated space of densities is determined by the undirected graph $\mathcal{G}$ which is chosen.

Alternative Problems with Additional Conditional Independence Structure

Consider now the set of all densities $p \in \mathcal{P}(V, d)$ which factor according to an arbitrary undirected graph $\mathcal{G} = (V, E)$. We denote this set by $\mathcal{P}_{\mathcal{G}}(V, d)$,[13] *i.e.* if $\mathcal{C}$ is the set of all maximal cliques of $\mathcal{G}$ and if $\psi_C(x_C)$ is any non-negative function defined on the variables x_C, then

$$\mathcal{P}_{\mathcal{G}}(V, d) \triangleq \left\{ p(x_V) \,\middle|\, p \in \mathcal{P}(V, d),\ p(x) = \prod_{C \in \mathcal{C}} \psi_C(x_C) \right\}. \tag{3.21}$$

Since any density $p \in \mathcal{P}_{\mathcal{G}}(V, d)$ factors according to $\mathcal{G}$, Theorem 3.2 indicates that a process X with such a density p is Markov with respect to $\mathcal{G}$, and therefore, by considering the set $\mathcal{P}_{\mathcal{G}}(V, d)$, we are focusing on processes with the conditional independence structure implied by $\mathcal{G}$. In what follows, we choose not to consider the entire set $\mathcal{P}_{\mathcal{G}}(V, d)$ but only the subset of densities $p(x)$ whose marginals $p(x_M)$ match a target density $p^*(x_M)$, and we denote this set by $\mathcal{P}_{\mathcal{G}}^M(V, d)$, *i.e.*

$$\mathcal{P}_{\mathcal{G}}^M(V, d) \triangleq \{ p(x_V) \,|\, p \in \mathcal{P}_{\mathcal{G}}(V, d),\ p(x_M) = p^*(x_M) \}. \tag{3.22}$$

Using the set $\mathcal{P}_{\mathcal{G}}^M(V, d)$, the following procedure, which we call *alternative problem* $\mathcal{P}_{\mathcal{G}}^M$, is a natural generalization of problem $\mathcal{P}^M$.

> <u>Alternative Problem $\mathcal{P}_{\mathcal{G}}^M$:</u> Find any density $\hat{p} \in \mathcal{P}_{\mathcal{G}}^M(V, d)$ such that $\hat{p}^T(x_M) = p^*(x_M)$.

Notice that by setting $\mathcal{G} = \mathcal{G}_{\preceq}^{\sim}$, the preceding problem is equivalent to problem $\mathcal{Q}$, and by letting $\mathcal{G}$ be the complete graph, the preceding problem is equivalent to problem $\mathcal{P}^M$. In addition to these special cases, problem $\mathcal{P}_{\mathcal{G}}^M$ suggests a range of different formulations for undirected graphs which lie "between" $\mathcal{G}_{\preceq}^{\sim}$ and the complete graph. As the following proposition indicates, problems $\mathcal{P}_{\mathcal{G}}^M$ and $\mathcal{Q}$ are compatible, as long as $\mathcal{G}$ is a supergraph of $\mathcal{G}_{\preceq}^{\sim}$.

[13] Notice that the set $\mathcal{P}_{\mathcal{G}_{\preceq}}(V, d)$ of all multiscale models is equivalent to the set $\mathcal{P}_{\mathcal{G}_{\preceq}^{\sim}}(V, d)$.

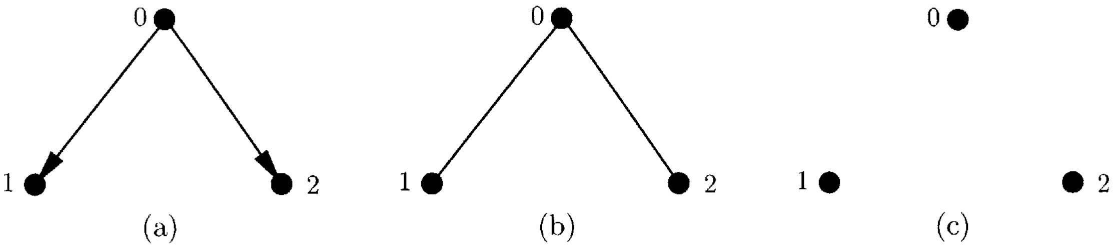


Figure 3.5. Graphs considered in Example 3.2. (a) Rooted tree $\mathcal{G}_{\preceq}$. (b) The undirected version $\mathcal{G}^{\sim}_{\preceq}$ of the rooted tree in (a). (c) A graph $\mathcal{G}$ which is not a supergraph of $\mathcal{G}^{\sim}_{\preceq}$.

Proposition 3.3 (Relationship Between Solutions to $\mathcal{P}_{\mathcal{G}}^{M}$ and $\mathcal{Q}$).
Let $\mathcal{G}_{\preceq}$ be a rooted tree defined on vertex set V, and let $p^*(x_M)$ be a given target density. If a graph $\mathcal{G} = (V, E)$ is a supergraph of $\mathcal{G}^{\sim}_{\preceq}$, the mapping $p \longrightarrow p^T$ is a surjection from the solution set of problem $\mathcal{P}_{\mathcal{G}}^{M}$ onto the solution set of problem $\mathcal{Q}$.

Proof. For every solution $\hat{q}$ of problem $\mathcal{Q}$, $\hat{q}$ is also a solution to problem $\mathcal{P}_{\mathcal{G}}^{M}$ since $\hat{q} \in \mathcal{P}_{\mathcal{G}}^{M}(V, d)$ (due to the fact that $\mathcal{G}$ is a supergraph of $\mathcal{G}^{\sim}_{\preceq}$). Hence, $q \longrightarrow q$ is a mapping from the solution set of $\mathcal{Q}$ to the solution set of $\mathcal{P}_{\mathcal{G}}^{M}$. In addition, $\hat{q}^T = \hat{q}$ which proves that the identity map is a right inverse of $p \longrightarrow p^T$ on the solution set of problem $\mathcal{Q}$ and thereby proves the proposition. ■

When $\mathcal{G}$ is not a supergraph of $\mathcal{G}^{\sim}_{\preceq}$, the mapping $p \longrightarrow p^T$ may be surjective, but this only occurs for special choices of the target density $p^*(x_M)$. When $p^*(x_M)$ fails to have special structure and when $\mathcal{G}$ is not a supergraph of $\mathcal{G}^{\sim}_{\preceq}$, the mapping $p \longrightarrow p^T$ is not surjective, and consequently, all solutions to problem $\mathcal{Q}$ cannot be identified from solutions to problem $\mathcal{P}_{\mathcal{G}}^{M}$. The following example provides a simple illustration of these ideas.

Example 3.2 (On the Compatibility of Problems $\mathcal{P}_{\mathcal{G}}^{M}$ and $\mathcal{Q}$).
Consider the three graphs shown in Figure 3.5. The rooted tree $\mathcal{G}_{\preceq}$ shown in Figure 3.5(a) indicates the structure of the multiscale model to be designed in this example, *i.e.* our ultimate goal is to solve problem $\mathcal{Q}$ for this choice of $\mathcal{G}_{\preceq}$. The undirected version $\mathcal{G}^{\sim}_{\preceq}$ of $\mathcal{G}_{\preceq}$ is shown in Figure 3.5(b), and a graph $\mathcal{G}$ which is not a supergraph of $\mathcal{G}^{\sim}_{\preceq}$ is shown in Figure 3.5(c). Consider solving problem $\mathcal{P}_{\mathcal{G}}^{M}$ for the graph $\mathcal{G}$ in Figure 3.5(c) and for a given target density $p^*(x_M)$ with $M = \{1, 2\}$.

Suppose the target density $p^*(x_M)$ has no independence structure – in particular, $p^*(x_1, x_2) \neq p^*(x_1)p^*(x_2)$. Any density p which factors according to $\mathcal{G}$ must satisfy $p(x) = p(x_0)p(x_1)p(x_2)$, implying that $p(x_1, x_2) = p(x_1)p(x_2)$. Since $p^*(x_1, x_2)$ does not possess this independence structure, the set $\mathcal{P}_{\mathcal{G}}^{M}(V, d)$ is empty, and therefore, problem $\mathcal{P}_{\mathcal{G}}^{M}$ has no solution. This restriction on $p^*(x_M)$ does not however suggest that problem $\mathcal{Q}$ has no solution, and consequently, if problem $\mathcal{Q}$ has a least one solution, it cannot be identified by solving problem $\mathcal{P}_{\mathcal{G}}^{M}$. This shows the importance of $\mathcal{G}$ being a supergraph of $\mathcal{G}^{\sim}_{\preceq}$; if this is not the case, problem $\mathcal{P}_{\mathcal{G}}^{M}$ searches over a space of densities with conditional independencies not possessed by the multiscale model of interest.

The independence structure of $p^*(x_M)$ alone does not indicate whether the mapping $p \longrightarrow p^T$ is surjective or not. Suppose now that $p^*(x_1, x_2) = p^*(x_1)p^*(x_2)$ and that problems $\mathcal{Q}$ and $\mathcal{P}_{\mathcal{G}}^{M}$ both have solutions. This example is special in the sense that every solution $\hat{p}$ of problem $\mathcal{P}_{\mathcal{G}}^{M}$ gets mapped to itself under $p \longrightarrow p^T$, *i.e.* since $\hat{p}(x) = \hat{p}(x_0)\hat{p}(x_1)\hat{p}(x_2)$, then $\hat{p}^T(x) = \hat{p}(x)$. This suggests that $p \longrightarrow p^T$ is a surjection if and only if every solution $\hat{q}$ of $\mathcal{Q}$ also factors according to $\mathcal{G}$.

However, just because a solution to $\mathcal{Q}$ satisfies $\hat{q}(x_1, x_2) = \hat{q}(x_1)\hat{q}(x_2) = p^*(x_1, x_2)$ does not imply that it satisfies $\hat{q}(x) = \hat{q}(x_0)\hat{q}(x_1)\hat{q}(x_2)$. Hence, the projection mapping may not be surjective even when $p^*(x_M)$ has independence structure.

There are special cases where the choice of $p^*(x_M)$ leads to a surjective mapping. As an example, suppose $X = (X_0, X_1, X_2)$ takes values in the space $\{0,1\} \times \{0,1\} \times \{0,1\}$, and suppose the goal is to match the following target density,

$$p^*(x_1, x_2) = \begin{cases} 1 & \text{if } x_1 = 1 \text{ and } x_2 = 1, \\ 0 & \text{otherwise.} \end{cases}$$

Since we constrain X_0 to be a binary random variable, the set $\mathcal{P}^M(V, d)$ only contains densities of the form,

$$p(x_0, x_1, x_2) = \begin{cases} p_0 & \text{if } x_0 = 0, x_1 = 1, \text{ and } x_2 = 1, \\ 1 - p_0 & \text{if } x_0 = 1, x_1 = 1, \text{ and } x_2 = 1, \\ 0 & \text{otherwise,} \end{cases}$$

where $p_0 \in [0, 1]$. Notice that all such densities $p \in \mathcal{P}^M(V, d)$ satisfy $p(x_0, x_1, x_2) = p(x_0)p(x_1)p(x_2)$. Consequently, the solution sets to problems $\mathcal{Q}$ and $\mathcal{P}_{\mathcal{G}}^M$ are identically equal to $\mathcal{P}^M(V, d)$, and $p \longrightarrow p^T$ is the identity mapping on this set. ◀

Since $p^*(x_M)$ must be a special density for the mapping $p \longrightarrow p^T$ to be surjective, we henceforth assume that $\mathcal{G}$ is a supergraph of $\mathcal{G}_{\preceq}^{\sim}$ when considering the alternative problem $\mathcal{P}_{\mathcal{G}}^M$.

Relating Solutions to Different Alternative Problems

In addition to the relationship between problems $\mathcal{P}_{\mathcal{G}}^M$ and $\mathcal{Q}$ given in Proposition 3.3, there exists an important relationship between the solutions to problems $\mathcal{P}_{\mathcal{G}}^M$ and $\mathcal{P}_{\mathcal{G}'}^M$ for some choices of $\mathcal{G}$ and $\mathcal{G}'$. In this section, we characterize the types of graphs $\mathcal{G}$ and $\mathcal{G}'$ for which an obvious surjection exists between the solution sets of $\mathcal{P}_{\mathcal{G}}^M$ and $\mathcal{P}_{\mathcal{G}'}^M$. In particular, we consider the projection mapping $p \longrightarrow p_{\mathcal{G}}$ introduced in Section 3.3.2, which is well-defined for all triangulated graphs $\mathcal{G}$ and is a generalization of the mapping $p \longrightarrow p^T$.

Let $\mathcal{G}$ and $\mathcal{G}'$ be supergraphs of a given tree $\mathcal{G}_{\preceq}^{\sim}$, and consider the corresponding problems $\mathcal{P}_{\mathcal{G}}^M$ and $\mathcal{P}_{\mathcal{G}'}^M$. We want to know what choices for $\mathcal{G}$ and $\mathcal{G}'$ guarantee that $p \longrightarrow p_{\mathcal{G}}$ is a surjective mapping from the solution set of problem $\mathcal{P}_{\mathcal{G}'}^M$ onto the solution set of problem $\mathcal{P}_{\mathcal{G}}^M$. There are three possibilities to consider:

(1) $\mathcal{G}$ is not a supergraph of $\mathcal{G}'$, and $\mathcal{G}'$ is not a supergraph of $\mathcal{G}$.

(2) $\mathcal{G}'$ is a supergraph of $\mathcal{G}$, and $\mathcal{G}$ is not triangulated.

(3) $\mathcal{G}'$ is a supergraph of $\mathcal{G}$, and $\mathcal{G}$ is triangulated.

As discussed below, $p \longrightarrow p_{\mathcal{G}}$ is only guaranteed to be a surjective mapping for a specific subset of the problems in the third case.

First, recall that the mapping $p \longrightarrow p_{\mathcal{G}}$ is not defined for non-triangulated graphs $\mathcal{G}$. As such, we can say nothing about the second case, and we can only consider triangulated graphs $\mathcal{G}$ in the first case. Suppose the first case holds for some triangulated graph $\mathcal{G}$, and consider any solution $\hat{p}$ to problem $\mathcal{P}_{\mathcal{G}'}^{M}$. Since $\hat{p}$ is an element of the set $\mathcal{P}_{\mathcal{G}'}^{M}(V,d)$, it satisfies the conditional independencies implied by the graph $\mathcal{G}'$. Likewise, consider the projected density $\hat{p}_{\mathcal{G}}$. Since $\hat{p}_{\mathcal{G}}$ maintains the marginals of $\hat{p}$ on the maximal cliques of $\mathcal{G}$, it satisfies the conditional independencies implied by both $\mathcal{G}$ and $\mathcal{G}'$. Consequently, if there is any solution to problem $\mathcal{P}_{\mathcal{G}}^{M}$ that does not satisfy the conditional independencies of both $\mathcal{G}$ and $\mathcal{G}'$, the mapping $p \longrightarrow p_{\mathcal{G}}$ is not a surjection. This situation is intuitively similar to the problem considered previously in Example 3.2.

Finally, consider the third case. Given the fact that $\mathcal{G}'$ is a supergraph of $\mathcal{G}$, it seems plausible that $p \longrightarrow p_{\mathcal{G}}$ is in fact the needed surjection, but as we discuss, if the graph $\mathcal{G}$ does not have a clique equal to M, a solution $\hat{p}$ of problem $\mathcal{P}_{\mathcal{G}'}^{M}$ is not guaranteed to map to a solution $\hat{p}_{\mathcal{G}}$ of problem $\mathcal{P}_{\mathcal{G}}^{M}$. Therefore, the projection operation $p \longrightarrow p_{\mathcal{G}}$ may not be a mapping from the solution set of problem $\mathcal{P}_{\mathcal{G}'}^{M}$ to the solution set of problem $\mathcal{P}_{\mathcal{G}}^{M}$ in this case.

Let $\hat{p}$ be any solution to problem $\mathcal{P}_{\mathcal{G}'}^{M}$. In order for $\hat{p}_{\mathcal{G}}$ to be a solution of problem $\mathcal{P}_{\mathcal{G}}^{M}$, it must satisfy two constraints:

(1) $(\hat{p}_{\mathcal{G}})^{T}(x_M) = p^{*}(x_M)$,

(2) $\hat{p}_{\mathcal{G}} \in \mathcal{P}_{\mathcal{G}}^{M}(V,d)$.

The first constraint is always satisfied, as evidenced by the following lemma.

Lemma 3.3 (Nested Projections).
Let $\mathcal{H}$ be any triangulated graph defined on vertex set V, and let $\mathcal{G} = (V,E)$ be any triangulated supergraph of $\mathcal{H}$. Given any density $p(x_V)$, the two densities $(p_{\mathcal{G}})_{\mathcal{H}}$ and $p_{\mathcal{H}}$ are equal, i.e. *projecting p onto the graph $\mathcal{H}$ is the same as first projecting p onto $\mathcal{G}$ and then $\mathcal{H}$.*

Proof. This follows directly from the definition of the projection operation and the fact that $\mathcal{G}$ is a triangulated supergraph of $\mathcal{H}$. Since $\mathcal{G}$ is a supergraph of $\mathcal{H}$, every maximal clique of $\mathcal{H}$ is contained in a maximal clique of $\mathcal{G}$. Consequently, the marginals $p(x_C)$ of $p_{\mathcal{G}}$ are identical to the marginals $p(x_C)$ of p for every maximal clique C of $\mathcal{H}$. ■

Setting $\mathcal{H} = \mathcal{G}_{\preceq}^{\sim}$ in the preceding lemma shows that $(\hat{p}_{\mathcal{G}})^{T} = \hat{p}^{T}$, and therefore, $(\hat{p}_{\mathcal{G}})^{T}(x_M) = \hat{p}^{T}(x_M) = p^{*}(x_M)$.

In order for the second constraint to be satisfied, $\hat{p}_{\mathcal{G}}$ must factor according to $\mathcal{G}$ (which is true by definition), and in addition, $\hat{p}_{\mathcal{G}}(x_M)$ must equal $p^{*}(x_M)$. It is the latter constraint $\hat{p}_{\mathcal{G}}(x_M) = p^{*}(x_M)$ which may not be satisfied when $\mathcal{G}$ has no clique equal to M, as the following example illustrates.

Example 3.3 (On the Compatibility of Problems $\mathcal{P}_{\mathcal{G}}^{M}$ and $\mathcal{P}_{\mathcal{G}'}^{M}$).
Consider the graphs $\mathcal{G}_{\preceq}$ and $\mathcal{G}_{\preceq}^{\sim}$ shown in Figures 3.6(a) and (b), which correspond to the multiscale model to be designed in this example. We are interested in the two alternative problems $\mathcal{P}_{\mathcal{G}}^{M}$ and $\mathcal{P}_{\mathcal{G}'}^{M}$ where $\mathcal{G}$ is shown in Figure 3.6(c) and $\mathcal{G}'$ is the complete graph on the vertices $V = \{0,1,2,3\}$. The purpose of this example is to show that the projection operation $p \longrightarrow p_{\mathcal{G}}$ is not necessarily a mapping from the solution set of problem $\mathcal{P}_{\mathcal{G}'}^{M}$ to the solution set of problem $\mathcal{P}_{\mathcal{G}}^{M}$, and we accomplish this by providing a specific solution $\hat{p}$ of problem $\mathcal{P}_{\mathcal{G}'}^{M}$ (or equivalently problem $\mathcal{P}^{M}$) which does not project to a solution $\hat{p}_{\mathcal{G}}$ of problem $\mathcal{P}_{\mathcal{G}}^{M}$.

In this example, we assume that X_0, X_1, X_2, X_3 are binary random variables, each taking values in the space $\{0,1\}$. Now, let $M = \{1,2,3\}$, and suppose the target density $p^{*}(x_M)$ is specified by the probabilities shown in Table 3.1, *i.e.* X_1, X_2, X_3 are uniformly distributed on $\{0,1\} \times \{0,1\} \times \{0,1\}$

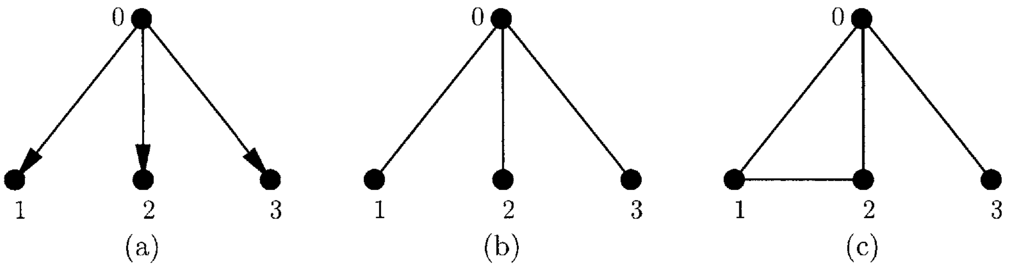


Figure 3.6. Graphs considered in Example 3.3. (a) Rooted tree $\mathcal{G}_{\preceq}$. (b) The undirected version $\mathcal{G}_{\preceq}^{\sim}$ of the rooted tree in (a). (c) A graph $\mathcal{G}$ which is a triangulated supergraph of $\mathcal{G}_{\preceq}^{\sim}$, but does not have a clique equal to $M = \{1, 2, 3\}$.

	$X_1 = 0,\ X_2 = 0$	$X_1 = 0,\ X_2 = 1$	$X_1 = 1,\ X_2 = 0$	$X_1 = 1,\ X_2 = 1$
$X_3 = 0$	1/8	1/8	1/8	1/8
$X_3 = 1$	1/8	1/8	1/8	1/8

Table 3.1: Table of probabilities for the target density $p^*(x_1, x_2, x_3)$ considered in Example 3.3.

	$X_1 = 0,\ X_2 = 0$	$X_1 = 0,\ X_2 = 1$	$X_1 = 1,\ X_2 = 0$	$X_1 = 1,\ X_2 = 1$
$X_3 = 0,\ X_0 = 0$	1/16	0	0	1/16
$X_3 = 0,\ X_0 = 1$	1/16	1/8	1/8	1/16
$X_3 = 1,\ X_0 = 0$	1/16	1/16	1/16	1/16
$X_3 = 1,\ X_0 = 1$	1/16	1/16	1/16	1/16

Table 3.2. Table of probabilities for the discrete density $\hat{p}(x_0, x_1, x_2, x_3)$ considered in Example 3.3. This density satisfies $\hat{p}(x_1, x_2, x_3) = \hat{p}^T(x_1, x_2, x_3)$ for the tree $\mathcal{G}_{\preceq}^{\sim}$ shown in Figure 3.6(b).

under the target density $p^*(x_M)$. Given this target density, we want to find a density $\hat{p}(x_0, x_1, x_2, x_3)$ such that $\hat{p}(x_M) = p^*(x_M) = \hat{p}^T(x_M)$, *i.e.* a solution $\hat{p}$ to problem $\mathcal{P}^M$. One possible solution $\hat{p}(x_0, x_1, x_2, x_3)$ is characterized by the values shown in Table 3.2. Marginalizing over variable X_0 in Table 3.2, one can see that $\hat{p}(x_1, x_2, x_3) = p^*(x_1, x_2, x_3)$.[14] While we do not show it here, it is also true that $\hat{p}^T(x_M) = p^*(x_M)$, and therefore, $\hat{p}^T$ is a solution to exact realization problem $\mathcal{Q}$.

Now, consider projecting $\hat{p}$ onto the graph $\mathcal{G}$ in Figure 3.6(c). As proven earlier in Lemma 3.3, the density $(\hat{p}_{\mathcal{G}})^T$ is equal to $\hat{p}^T$, and therefore, $(\hat{p}_{\mathcal{G}})^T(x_M) = p^*(x_M)$. This means that the density $\hat{p}$ satisfies three very important equalities, namely $\hat{p}(x_M) = (\hat{p}_{\mathcal{G}})^T(x_M) = \hat{p}^T(x_M) = p^*(x_M)$. These equalities place very strong constraints on the density $\hat{p}$, and in this example, we want to know if these constraints necessarily imply the equality $\hat{p}_{\mathcal{G}}(x_M) = p^*(x_M)$. The answer is no, however, since the marginal $\hat{p}_{\mathcal{G}}(x_M)$ is not uniformly distributed on $\{0,1\} \times \{0,1\} \times \{0,1\}$, as indicated by the probabilities in Table 3.3. This indicates that $\hat{p}_{\mathcal{G}}$ is not a solution to problem $\mathcal{P}_{\mathcal{G}}^M$, and consequently, $p \longrightarrow p_{\mathcal{G}}$ is not a mapping from the solution set of $\mathcal{P}^M$ to the solution set of $\mathcal{P}_{\mathcal{G}}^M$. ◀

[14] The discrete probability table for $\hat{p}(x_1, x_2, x_3)$ can be obtained by simply adding the appropriate entries of Table 3.2. The values for $\hat{p}(X_1 = x_1, X_2 = x_2, X_3 = 0)$ are obtained by adding the first two rows of Table 3.2, while the values for $\hat{p}(X_1 = x_1, X_2 = x_2, X_3 = 1)$ are obtained by adding the last two rows of Table 3.2.

	$X_1 = 0,\, X_2 = 0$	$X_1 = 0,\, X_2 = 1$	$X_1 = 1,\, X_2 = 0$	$X_1 = 1,\, X_2 = 1$
$X_3 = 0$	7/60	8/60	8/60	7/60
$X_3 = 1$	8/60	7/60	7/60	8/60

Table 3.3. Table of probabilities for the marginal density $\hat{p}_{\mathcal{G}}(x_1, x_2, x_3)$ considered in Example 3.3, where $\hat{p}(x_0, x_1, x_2, x_3)$ is specified by the values in Table 3.2 and where the graph $\mathcal{G}$ is shown in Figure 3.6(c).

When a triangulated graph $\mathcal{G}$ has a clique equal to M, two densities p and $p_{\mathcal{G}}$ are by definition guaranteed to have the same marginal on the variables x_M. As a consequence of this fact and Lemma 3.3, the projection operation $p \longrightarrow p_{\mathcal{G}}$ is in this case a mapping from the solution set of problem $\mathcal{P}_{\mathcal{G}'}^{M}$ to the solution set of problem $\mathcal{P}_{\mathcal{G}}^{M}$, and as the following proposition indicates, the mapping is also a surjection.

Proposition 3.4 (Relationship Between Solutions to $\mathcal{P}_{\mathcal{G}}^{M}$ and $\mathcal{P}_{\mathcal{G}'}^{M}$).
Let $\mathcal{G}_{\preceq}$ be a rooted tree defined on vertex set V, and let $p^*(x_M)$ be a given target density. If graph $\mathcal{G} = (V, E)$ is a triangulated supergraph of $\mathcal{G}_{\preceq}^{\sim}$ with a clique equal to M and if $\mathcal{G}' = (V, E')$ is a supergraph of $\mathcal{G}$, then the mapping $p \longrightarrow p_{\mathcal{G}}$ is a surjection from the solution set of problem $\mathcal{P}_{\mathcal{G}'}^{M}$ onto the solution set of problem $\mathcal{P}_{\mathcal{G}}^{M}$.

Proof. From the preceding discussion, $p \longrightarrow p_{\mathcal{G}}$ is in this case a mapping from the solution set of problem $\mathcal{P}_{\mathcal{G}'}^{M}$ to the solution set of problem $\mathcal{P}_{\mathcal{G}}^{M}$. Now, for every solution $\hat{q}$ of problem $\mathcal{P}_{\mathcal{G}}^{M}$, $\hat{q}$ is also a solution to problem $\mathcal{P}_{\mathcal{G}'}^{M}$ since $\hat{q} \in \mathcal{P}_{\mathcal{G}'}^{M}(V, d)$ (due to the fact that $\mathcal{G}'$ is a supergraph of $\mathcal{G}$). Hence, $q \longrightarrow q$ is a mapping from the solution set of $\mathcal{P}_{\mathcal{G}}^{M}$ to the solution set of $\mathcal{P}_{\mathcal{G}'}^{M}$. In addition, $\hat{q}_{\mathcal{G}} = \hat{q}$ proving that the identity map is a right inverse of $p \longrightarrow p_{\mathcal{G}}$ and thereby proving the proposition. ■

■ 3.4.3 Sufficient Conditions for Exact Realization

In the previous section, the relationship between solutions to problems $\mathcal{Q}$, $\mathcal{P}_{\mathcal{G}}^{M}$, and $\mathcal{P}_{\mathcal{G}'}^{M}$ was examined. In this section, this relationship is used to suggest sufficient conditions for solutions to problem $\mathcal{P}^{M}$, which in turn leads to solutions to problem $\mathcal{Q}$ via the mapping $p \longrightarrow p^{T}$. As we show, the sufficient conditions suggested here are less stringent than the two constraints $\hat{p} \in \mathcal{P}^{M}(V, d)$ and $\hat{p} = \hat{p}^{T}$ introduced in Section 3.4.1. Before stating these conditions, though, we first consider sufficient conditions for the more general problem $\mathcal{P}_{\mathcal{G}}^{M}$.

The sufficient conditions $\hat{p} \in \mathcal{P}^{M}(V, d)$ and $\hat{p} = \hat{p}^{T}$ for problem $\mathcal{P}^{M}$ easily generalize to sufficient conditions for the more general problem $\mathcal{P}_{\mathcal{G}}^{M}$. Specifically, suppose a density $\hat{p}$ satisfies the following two constraints:

$$\underline{\text{Condition 1}} \quad \hat{p} \in \mathcal{P}_{\mathcal{G}}^{M}(V, d),$$

$$\underline{\text{Condition 2}} \quad \hat{p} = \hat{p}^{T}.$$

Then, $\hat{p}$ is a solution to problem $\mathcal{P}_{\mathcal{G}}^{M}$ since the preceding conditions imply $\hat{p}(x_M) = \hat{p}^{T}(x_M) = p^*(x_M)$.

While these sufficient conditions are simple to state, they are not necessarily simple to use in a practical sense. In particular, depending on the choice of $\mathcal{G}$, it may be difficult to satisfy the first constraint, satisfy the second constraint, or simultaneously satisfy both constraints. The first constraint is challenging when the set $\mathcal{P}_{\mathcal{G}}^{M}(V,d)$ is difficult to characterize, *i.e.* for some choices of $\mathcal{G}$, it can be challenging to find a density $\hat{p}$ which lies in the set $\mathcal{P}_{\mathcal{G}}^{M}(V,d)$. On the other hand, the second constraint $\hat{p} = \hat{p}^T$ is difficult to satisfy when the set $\mathcal{P}_{\mathcal{G}}^{M}(V,d)$ is large, *i.e.* if $\mathcal{P}_{\mathcal{G}}^{M}(V,d)$ contains a large number of densities which do not satisfy $\hat{p} = \hat{p}^T$, then searching for a density which does satisfy this constraint may be computationally prohibitive.

As a specific illustration of the preceding discussion, consider the two possible extremes for $\mathcal{G}$, *i.e.* when $\mathcal{G}$ is equal to $\mathcal{G}_{\preceq}^{\sim}$ and when $\mathcal{G}$ is the complete graph. When $\mathcal{G} = \mathcal{G}_{\preceq}^{\sim}$, characterizing the set $\mathcal{P}_{\mathcal{G}_{\preceq}^{\sim}}^{M}(V,d)$ is equivalent to finding every solution to problem $\mathcal{Q}$, and if we could perform this task, we would simply solve the original realization problem. In addition, notice that every density $\hat{p} \in \mathcal{P}_{\mathcal{G}_{\preceq}^{\sim}}^{M}(V,d)$ is a multiscale model and therefore satisfies $\hat{p} = \hat{p}^T$. Consequently, the second constraint is trivially satisfied in this case. At the other extreme, when $\mathcal{G}$ is the complete graph, the set $\mathcal{P}^{M}(V,d)$ is easy to characterize since it consists of all densities of the form $p(x) = p(x_{V-M}|x_M)p^*(x_M)$, *i.e.* choose any conditional density $p(x_{V-M}|x_M)$ such that $p(x) = p(x_{V-M}|x_M)p^*(x_M)$ is consistent with dimensions d. On the other hand, satisfying the second constraint is a non-trivial task, since finding a density $\hat{p} = \hat{p}^T$ constitutes the large search necessary to solve problem $\mathcal{P}^M$.

For graphs $\mathcal{G}$ which lie between $\mathcal{G}_{\preceq}^{\sim}$ and the complete graph, there is a tradeoff in the ease of satisfying both of these constraints. For some choices of $\mathcal{G}$, the first condition is easier to satisfy than the second and *vice-versa*. In the remainder of this section, we focus on a particular subset of graphs $\mathcal{G}$ which somewhat mediate this tradeoff – namely, triangulated supergraphs of $\mathcal{G}_{\preceq}^{\sim}$ which have a clique equal to M. These are precisely the graphs $\mathcal{G}$ considered previously in Proposition 3.4.

For such a graph $\mathcal{G}$, the set $\mathcal{P}_{\mathcal{G}}^{M}(V,d)$ can be completely characterized in terms of the set $\mathcal{P}^{M}(V,d)$ as follows,

$$\mathcal{P}_{\mathcal{G}}^{M}(V,d) = \left\{q(x)\,\middle|\,q = p_{\mathcal{G}}, \ \text{ for some } \ p \in \mathcal{P}^{M}(V,d)\right\}. \tag{3.23}$$

In other words, $\mathcal{P}_{\mathcal{G}}^{M}(V,d)$ is the image of $\mathcal{P}^{M}(V,d)$ under the mapping $p \longrightarrow p_{\mathcal{G}}$, a fact which is a direct consequence of requiring $\mathcal{G}$ to have a clique equal to M.[15] Therefore, for this particular choice of $\mathcal{G}$, it is relatively simple to find a density that satisfies the first constraint. At the same time, since the set $\mathcal{P}_{\mathcal{G}}^{M}(V,d)$ has fewer elements than $\mathcal{P}^{M}(V,d)$ and since the elements of $\mathcal{P}_{\mathcal{G}}^{M}(V,d)$ possess additional factorization structure not possessed by all of the elements of $\mathcal{P}^{M}(V,d)$, the task of finding a density p which satisfies $p = p^T$ is relatively easier in the set $\mathcal{P}_{\mathcal{G}}^{M}(V,d)$ than $\mathcal{P}^{M}(V,d)$. Therefore, this particular choice of $\mathcal{G}$ provides a reasonable balance to the tradeoff between the two constraints.

In addition to the tradeoff, this choice of $\mathcal{G}$, along with Proposition 3.4, suggests interesting sufficient conditions for solutions to problem $\mathcal{P}^M$. In particular, suppose $\hat{q}$ satisfies the conditions $\hat{q} \in \mathcal{P}_{\mathcal{G}}^{M}(V,d)$ and $\hat{q} = \hat{q}^T$, *i.e.* the previously stated sufficient conditions for problem $\mathcal{P}_{\mathcal{G}}^{M}$, so that $\hat{q}$ is a solution to problem $\mathcal{P}_{\mathcal{G}}^{M}$. Using Proposition 3.4, any density $\hat{p} \in \mathcal{P}^{M}(V,d)$ which satisfies $\hat{q} = \hat{p}_{\mathcal{G}}$ is therefore a solution to problem $\mathcal{P}^{M}(V,d)$, and using Lemma 3.3, all of these conditions can be stated directly in terms of $\hat{p}$, *i.e.* $\hat{p} \in \mathcal{P}^{M}(V,d)$ and $\hat{p}_{\mathcal{G}} = (\hat{p}_{\mathcal{G}})^T = \hat{p}^T$. Hence, we have used the fact that $p \longrightarrow p_{\mathcal{G}}$ is a surjection in order to translate the sufficient conditions for problem $\mathcal{P}_{\mathcal{G}}^{M}$ directly into sufficient conditions for problem $\mathcal{P}^{M}$, and these new conditions depend on the

[15] If $\mathcal{G}$ is triangulated but has no clique equal to M, then characterizing $\mathcal{P}_{\mathcal{G}}^{M}(V,d)$ is more challenging, since the image of $\mathcal{P}^{M}(V,d)$ under the projection operation $p \longrightarrow p_{\mathcal{G}}$ is a superset of $\mathcal{P}_{\mathcal{G}}^{M}(V,d)$ in this case.

particular choice of $\mathcal{G}$. The following theorem summarizes this result.

Theorem 3.3 (Sufficient Conditions for Problem $\mathcal{P}^M$).
Let $\mathcal{G}_{\preceq}$ be a rooted tree defined on vertex set V, and let $p^*(x_M)$ be a given target density. Suppose $\mathcal{G} = (V, E)$ is any triangulated supergraph of $\mathcal{G}_{\preceq}^{\sim}$ with a clique equal to M. If a density $\hat{p} \in \mathcal{P}^M(V, d)$ satisfies $\hat{p}_{\mathcal{G}} = \hat{p}^T$, then $\hat{p}$ is a solution to problem $\mathcal{P}^M$, and $\hat{p}^T$ is a solution to problem $\mathcal{Q}$.

Proof. Follows directly from the sufficient conditions for solutions to problem $\mathcal{P}^M_{\mathcal{G}}$, Proposition 3.4, and Lemma 3.3. See the preceding discussion. $\blacksquare$

Notice that the sufficient conditions provided in Theorem 3.3 are less stringent than the two conditions $\hat{p} \in \mathcal{P}^M(V, d)$ and $\hat{p} = \hat{p}^T$ introduced in Section 3.4.1. In particular, the condition $\hat{p}_{\mathcal{G}} = \hat{p}^T$ requires that $\hat{p}$ match $\hat{p}^T$ only on the marginals indexed by the maximal cliques of $\mathcal{G}$, whereas the condition $\hat{p} = \hat{p}^T$ requires the entire density $\hat{p}$ to factor according to $\mathcal{G}_{\preceq}^{\sim}$. With this fact in mind, we would ideally like to minimize the number of edges in the graph $\mathcal{G}$, or in other words, choose a *sparse graph* since this would lead to less-stringent sufficient conditions. Of course, the degree of sparsity is limited by the fact that $\mathcal{G}$ must be a triangulated supergraph of $\mathcal{G}_{\preceq}^{\sim}$ and have a clique equal to M in order for $\hat{p}_{\mathcal{G}} = \hat{p}^T$ to be a sufficient condition. Therefore, we would ultimately like to choose the sparsest graph $\mathcal{G}$ which satisfies the conditions in Theorem 3.3.

While Theorem 3.3 provides a very succinct sufficient condition for solutions to problem $\mathcal{P}^M$, this condition is deceptively simple. In Sections 3.5– 3.8, we provide a graph-theoretic framework to facilitate a deeper understanding of this important result, and we ultimately show in Section 3.9.1 that Theorem 3.3 is equivalent to the result provided in Theorem 2.3 concerning marginalization-invariant Markovianity.

■ 3.4.4 Sufficient Conditions For Exact Realization with Augmented States

Using the notion of an augmented graph along with a few minor alterations in the results provided in Sections 3.4.2 and 3.4.3, the sufficient conditions stated in Theorem 3.3 can be generalized to the realization problem where augmented states are allowed. Suppose a rooted tree $\mathcal{G}_{\preceq} = (V, E)$ and a target density $p^*(x_M)$ are specified.[16] Recall from Section 3.3.3 that $\mathcal{G}^{\sharp}_{\preceq}$ is the augmented graph defined in terms of $\mathcal{G}_{\preceq}$ and M which incorporates the two types of vectors $v^{(t)}$ and $v^{(d)}$. Given that $\mathcal{G}^{\sharp}_{\preceq}$ is defined on an expanded vertex set $V^{\sharp}$ (according to the rules discussed in Section 3.3.3), we redefine the set M as follows,[17]

$$M \triangleq \{v^{(t)} | v^{(t)} \in V^{\sharp}\}, \tag{3.24}$$

and we map the previous target density $p^*(x_M)$ to the new target density $p^*(x_M)$ which is now indexed by the set of all target vertices $v^{(t)}$ in the graph $\mathcal{G}^{\sharp}_{\preceq}$.

By redefining the set M, the statements of problems $\mathcal{P}^M$ and $\mathcal{Q}$ remain unchanged even when considering augmented states. That is, the goal of problem $\mathcal{P}^M$ is to find a density $\hat{p} \in \mathcal{P}^M(V, d)$ such that $\hat{p}^T(x_M) = p^*(x_M)$, and the goal of problem $\mathcal{Q}$ is to find a density $\hat{q} \in \mathcal{P}^M_{\mathcal{G}_{\preceq}}(V, d)$ such that $\hat{q}(x_M) = p^*(x_M)$. The only difference is that the marginals $\hat{p}^T(x_M)$, $\hat{q}(x_M)$, and $p^*(x_M)$, are now indexed by the target vertices $v^{(t)} \in V^{\sharp}$ rather than the vertices $v \in V$.

[16] We assume that the set M contains some non-leaf vertices; otherwise, augmented states are unnecessary.

[17] This redefined set M is equivalent to the augmented marginalization constraint set $M^{\sharp}$ defined in Section 2.7.3 via the "augmentation rule".

When using augmented states, the graph-theoretic aspects of the results in Sections 3.4.2 and 3.4.3 become slightly more interesting. This is due to the fact that we are now working on an expanded set of vertices, and in particular, the tree $\mathcal{G}_{\preceq}^{\sim}$ has been replaced by a more complicated triangulated graph $\mathcal{G}_{\preceq}^{\sharp}$. As discussed in Section 3.3.3, though, the two graphs $\mathcal{G}_{\preceq}^{\sim}$ and $\mathcal{G}_{\preceq}^{\sharp}$ imply the same conditional independencies, and therefore, the two projections $p_{\mathcal{G}_{\preceq}^{\sim}}$ and $p_{\mathcal{G}_{\preceq}^{\sharp}}$ can be considered interchangeably. In addition, the results provided in Proposition 3.3, Proposition 3.4, and Theorem 3.3 do not rely on the fact that $\mathcal{G}_{\preceq}^{\sim}$ is a tree, but instead, hold more generally for the class of triangulated graphs.

Rather than restate all of the preceding results for the case of augmented graphs, we simply restate the most important result, namely Theorem 3.3.

Theorem 3.4 (Sufficient Conditions for Problem $\mathcal{P}^M$ with Augmented States).
Let $\mathcal{G}_{\prec}$ be a rooted tree defined on vertex set V, and let $p^*(x_M)$ be a given target density. Let M be redefined according to (3.24), and let $p^*(x_M)$ be indexed according to this new set M. Suppose $\mathcal{G}^{\sharp} = (V^{\sharp}, E)$ is a triangulated supergraph of $\mathcal{G}_{\preceq}^{\sharp} = (V^{\sharp}, E^{\sharp})$ with a clique equal to M. If a density $\hat{p} \in \mathcal{P}^M(V,d)$ satisfies $\hat{p}_{\mathcal{G}^{\sharp}} = \hat{p}^T$, then $\hat{p}$ is a solution to problem $\mathcal{P}^M$, and $\hat{p}^T$ is a solution to problem $\mathcal{Q}$.

Proof. Since $\hat{p} \in \mathcal{P}^M(V,d)$ and $\mathcal{G}^{\sharp}$ has a clique equal to M, we have $\hat{p}_{\mathcal{G}^{\sharp}}(x_M) = \hat{p}(x_M) = p^*(x_M)$, and since $\hat{p}_{\mathcal{G}^{\sharp}} = \hat{p}^T$, we have $\hat{p}^T(x_M) = \hat{p}_{\mathcal{G}^{\sharp}}(x_M) = p^*(x_M)$, thereby proving that $\hat{p}$ is a solution to problem $\mathcal{P}^M$. ■

The essential difference between Theorem 3.4 and Theorem 3.3 is that all of the graph-theoretic requirements are stated in terms of augmented graphs – in particular, the augmented graph $\mathcal{G}^{\sharp}$ must be a supergraph of $\mathcal{G}_{\preceq}^{\sharp}$ and have a clique equal to the redefined set M. In Section 3.9.2, we show that Theorem 3.4 is equivalent to the result provided in Theorem 2.4 concerning marginalization-invariant Markovianity for augmented states.

3.5 A Road Map

In this section, we provide a motivating example to help guide the reader through the theory developed in Sections 3.6–3.8. In these subsequent sections, the graph-theoretic and probabilistic results apply to general triangulated graphs, and the ideas are presented generically without motivation for or reference to the exact realization problem. As such, the reader may need to occasionally refer back to this example in order to better understand the relationship between subsequent results and the realization problem. After establishing the needed theoretical results in Sections 3.6–3.8, we return to the exact realization problem in Section 3.9.

Example 3.4 (A Motivating Example).
Consider the rooted tree $\mathcal{G}_{\prec}$ and its corresponding undirected version $\mathcal{G}_{\preceq}^{\sim}$ shown previously in Figures 3.4(a) and (b), and suppose a target density $p^*(x_M)$ is indexed by the leaf vertices of $\mathcal{G}_{\prec}$, *i.e.* $M = \{3,4,5,6\}$.[18] Using Theorem 3.3, any density $p \in \mathcal{P}^M(V,d)$ satisfying the condition $p_{\mathcal{G}} = p^T$ (for an appropriate choice of $\mathcal{G}$) is a solution to problem $\mathcal{P}^M$. In this example, we study the constraints imposed on a density $p \in \mathcal{P}^M(V,d)$ by the condition $p_{\mathcal{G}} = p^T$; in particular, we provide a set of conditional independencies which ensure this constraint is satisfied.

[18] Recall that this is the same problem considered in Example 2.5 in Section 2.6.

Let X be a random process with density $p \in \mathcal{P}^M(V,d)$, and consider the following family of sets,

$$\begin{aligned}\mathcal{M}_0 &= \{\{0,3,4\},\{0,5,6\}\}\\ \mathcal{M}_1 &= \{\{0,1\},\{1,3\},\{1,4\}\}\\ \mathcal{M}_2 &= \{\{0,2\},\{2,5\},\{2,6\}\},\end{aligned}$$

along with the following equivalences,[19]

$$\perp X_{\mathcal{M}_0} \iff \begin{aligned} p(x_0,x_3,x_4,x_5,x_6) &= p(x_0)p(x_3,x_4|x_0)p(x_5,x_6|x_0)\\ &= \frac{p(x_0,x_3,x_4)p(x_0,x_5,x_6)}{p(x_0)}\end{aligned} \tag{3.25a}$$

$$\perp X_{\mathcal{M}_1} \iff \begin{aligned} p(x_0,x_1,x_3,x_4) &= p(x_1)p(x_0|x_1)p(x_3|x_1)p(x_4|x_1)\\ &= \frac{p(x_0,x_1)p(x_1,x_3)p(x_1,x_4)}{p(x_1)p(x_1)}\end{aligned} \tag{3.25b}$$

$$\perp X_{\mathcal{M}_2} \iff \begin{aligned} p(x_0,x_2,x_5,x_6) &= p(x_2)p(x_0|x_2)p(x_5|x_2)p(x_6|x_2)\\ &= \frac{p(x_0,x_2)p(x_2,x_5)p(x_2,x_6)}{p(x_2)p(x_2)}.\end{aligned} \tag{3.25c}$$

The first equivalence requires random vectors $\{X_3,X_4\}$ and $\{X_5,X_6\}$ to be conditionally independent given $X_0 = x_0$. The second equivalence requires X_0, X_3, and X_4 to be jointly independent conditioned on $X_1 = x_1$, and the third equivalence requires X_0, X_5, and X_6 to be jointly independent conditioned on $X_2 = x_2$.

If X satisfies the conditions $\perp X_{\mathcal{M}_0}$, $\perp X_{\mathcal{M}_1}$, and $\perp X_{\mathcal{M}_2}$, it can be shown that $p_{\mathcal{G}} = p^T$ for a specific graph $\mathcal{G}$. To do this, consider the following special sequence of densities defined in terms of particular marginals of p,

$$\begin{aligned}
q^{(0)}(x) = p^T(x) &= \frac{p(x_0,x_1)p(x_0,x_2)p(x_1,x_3)p(x_1,x_4)p(x_2,x_5)p(x_2,x_6)}{p(x_0)p(x_1)p(x_1)p(x_2)p(x_2)}\\
q^{(1)}(x) = q^{(0)}(x)\frac{p(x_0,x_2,x_5,x_6)}{\left[\dfrac{p(x_0,x_2)p(x_2,x_5)p(x_2,x_6)}{p(x_2)p(x_2)}\right]} &= \frac{p(x_0,x_1)p(x_1,x_3)p(x_1,x_4)p(x_0,x_2,x_5,x_6)}{p(x_0)p(x_1)p(x_1)}\\
q^{(2)}(x) = q^{(1)}(x)\frac{p(x_0,x_1,x_3,x_4)}{\left[\dfrac{p(x_0,x_1)p(x_1,x_3)p(x_1,x_4)}{p(x_1)p(x_1)}\right]} &= \frac{p(x_0,x_1,x_3,x_4)p(x_0,x_2,x_5,x_6)}{p(x_0)}\\
q^{(3)}(x) = q^{(2)}(x)\frac{p(x_0,x_3,x_4,x_5,x_6)}{\left[\dfrac{p(x_0,x_3,x_4)p(x_0,x_5,x_6)}{p(x_0)}\right]} &= \frac{p(x_0,x_1,x_3,x_4)p(x_0,x_2,x_5,x_6)p(x_0,x_3,x_4,x_5,x_6)}{p(x_0,x_3,x_4)p(x_0,x_5,x_6)}.
\end{aligned}$$

If X satisfies the conditions $\perp X_{\mathcal{M}_0}$, $\perp X_{\mathcal{M}_1}$, and $\perp X_{\mathcal{M}_2}$, the equivalences in (3.25) imply $q^{(0)} = q^{(1)} = q^{(2)} = q^{(3)}$.

[19] Recall from Definition 2.5 that the notation $\perp X_S$ is a shorthand for a specific set of conditional independencies.

The sequence of densities $q^{(0)}$, $q^{(1)}$, $q^{(2)}$, and $q^{(3)}$ is rather special. Specifically, each $q^{(i)}$ is the projection of p onto the graph $\mathcal{G}_i$, where $\mathcal{G}_0$, $\mathcal{G}_1$, $\mathcal{G}_2$, and $\mathcal{G}_3$, along with corresponding junction trees, are shown in Figure 3.7. Notice that the graphs $\mathcal{G}_i$, $i = 1, 2, 3$, are all triangulated supergraphs of $\mathcal{G}^{\sim}_{\preceq}$, and furthermore, $\mathcal{G}_3$ contains a clique equal to $M = \{3, 4, 5, 6\}$. Consequently, $\mathcal{G}_3$ fulfills the requirements of Theorem 3.3. Then, if X satisfies the conditions $\perp X_{\mathcal{M}_0}$, $\perp X_{\mathcal{M}_1}$, and $\perp X_{\mathcal{M}_2}$ so that $q^{(0)} = p^T = q^{(3)} = p_{\mathcal{G}_3}$, Theorem 3.3 indicates that p is a solution to problem $\mathcal{P}^M$.

As a final note, notice that the conditions $\perp X_{\mathcal{M}_0}$, $\perp X_{\mathcal{M}_1}$, and $\perp X_{\mathcal{M}_2}$ are the same as those previously considered in Example 2.5. In fact, these are the conditions required by the marginalization-invariant Markov property for the ordering $(0, 1, 2)$. This observation is not a coincidence, as we later show in Section 3.9. ◀

The preceding example suggests a list of conditional independencies that guarantee a solution to problem $\mathcal{P}^M$; of course, there are several other equally valid lists. The goal of Sections 3.6–3.8 is to: (1) develop a graph-theoretic framework by which to enumerate other possible lists and (2) derive an accompanying set of probabilistic relationships which prove that each such list implies the sufficient conditions in Theorem 3.3. In order fulfill this goal, there are three main issues to be addressed, each of which is highlighted by the preceding example:

(1) characterize the equivalences in (3.25) for more general problems,

(2) generate an appropriate sequence of graphs $\mathcal{G}_i$ in order to define the densities $q^{(i)} = p_{\mathcal{G}_i}$,

(3) derive the functional relationship that exists between $q^{(i)}$ and $q^{(i-1)}$.

We now consider each of these issues in turn.

Within the context of Example 3.4, the first issue is somewhat trivial since the equivalences in (3.25) hold by the definition of conditional independence. For more general problems, however, stating these equivalences succinctly is more challenging, and consequently, to deal with this particular issue, we introduce in Section 3.7.2 a graph-theoretic object called a *neighborhood separator*. The defining properties of a neighborhood separator allow it to be easily identified in a graph $\mathcal{G}$, and this type of object is directly linked to equivalences like those in (3.25). While we do not discuss the specific details of a neighborhood separator here, the interested reader should refer to Section 3.7.2 for its definition and application.

As Example 3.4 indicates, the sequence of graphs $\mathcal{G}_i$ shown in Figure 3.7 is special in that each $\mathcal{G}_i$ is a triangulated supergraph of $\mathcal{G}^{\sim}_{\preceq}$, and in addition, the final graph in the sequence has a clique equal to M. In Section 3.8, we suggest a method for generating such a sequence of graphs, using what we call the *modified elimination game* – a generalization of the *elimination game* discussed in Section 3.6. While the elimination game can be used to generate a sequence $\mathcal{G}_0, \ldots, \mathcal{G}_n$ of triangulated graphs, the graph $\mathcal{G}_n$ does not necessarily have a clique equal to M, and because of this fact, we introduce the modified elimination game.

Given a sequence of graphs $\mathcal{G}_0, \ldots, \mathcal{G}_n$ generated by the modified elimination game, the third and final issue concerns the functional relationship that exists between the densities $p_{\mathcal{G}_0}, \ldots, p_{\mathcal{G}_n}$. To address this issue, we show that the modified elimination game generates a sequence of *clique extensions*, *i.e.* each graph $\mathcal{G}_i$ has only one additional maximal clique not contained in $\mathcal{G}_{i-1}$. For example, the graphs shown in Figure 3.7 form such a sequence, since $\{0, 2, 5, 6\}$, $\{0, 1, 3, 4\}$, and $\{0, 3, 4, 5, 6\}$ are the unique new maximal cliques respectively contained in $\mathcal{G}_1$, $\mathcal{G}_2$, and $\mathcal{G}_3$. Since the modified elimination game generates a sequence of clique extensions, a simple functional relationship exists between $p_{\mathcal{G}_i}$ and $p_{\mathcal{G}_{i-1}}$ as shown in Section 3.7.1.

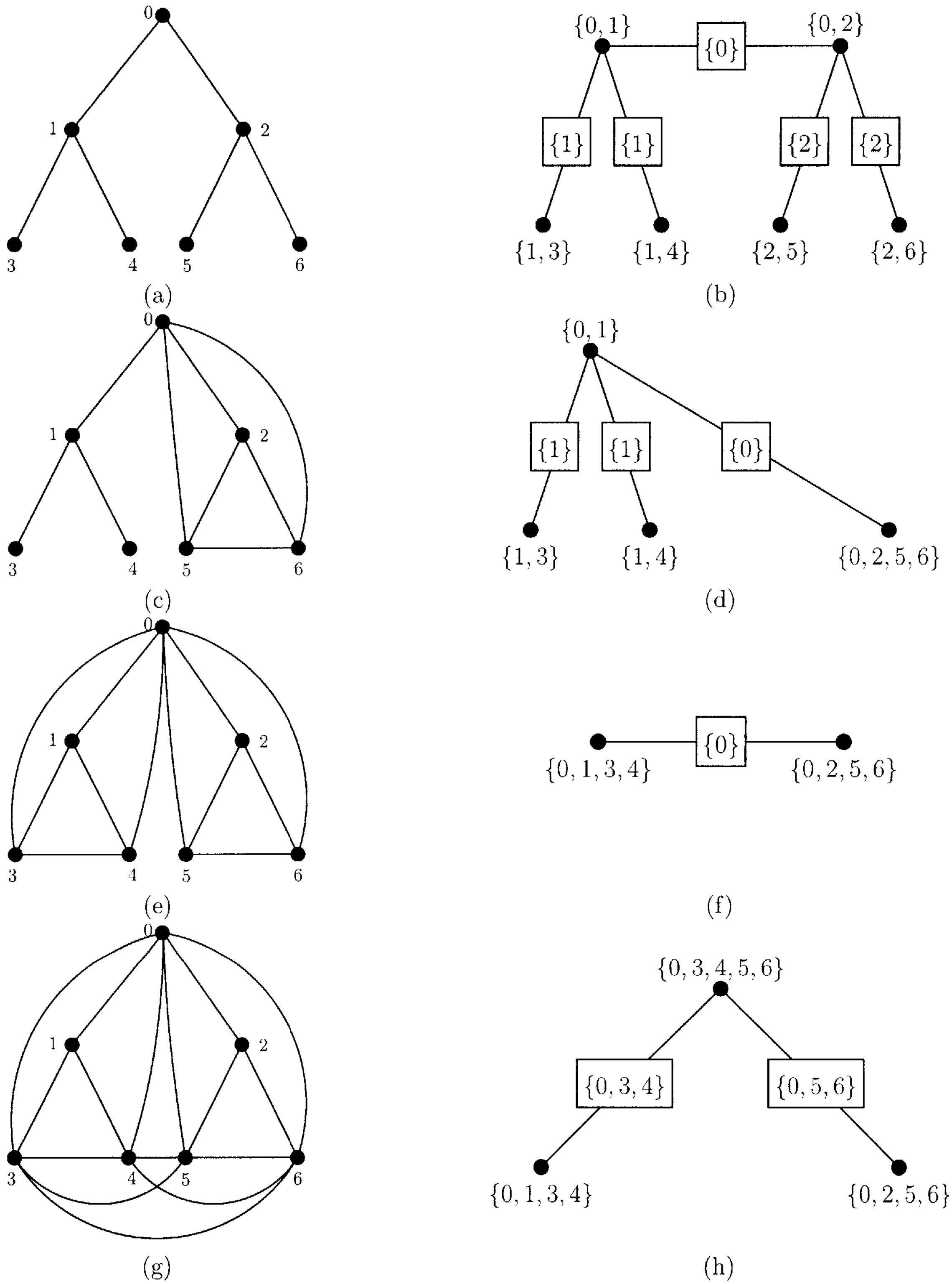


Figure 3.7. The sequence of triangulated graphs $\mathcal{G}_i$ and corresponding junction trees considered in Example 3.4. (a) $\mathcal{G}_0$ (b) A junction tree for $\mathcal{G}_0$. (c) $\mathcal{G}_1$ (d) A junction tree for $\mathcal{G}_1$. (e) $\mathcal{G}_2$ (f) A junction tree for $\mathcal{G}_2$. (g) $\mathcal{G}_3$ (h) A junction tree for $\mathcal{G}_3$.

3.6 The Elimination Game

The *elimination game* was first introduced by Parter [83] for the purpose of examining the fill generated by Gaussian elimination when solving a system of linear equations. Later, Rose [89] generalized the results of Parter and showed that there exist elimination sequences which produce no fill for a broader class of problems. We are interested in the elimination game solely for the purpose of introducing several graph-theoretic results and to further explore the relationship between graphs and probabilistic modeling. In this section, we formally define what we mean by the elimination game; we discuss the elimination game in the context of triangulated graphs; and we derive an important probabilistic relationship.

3.6.1 Definition and Notation

The key graphical operation involved in the elimination game is that of *vertex elimination*. For the immediate discussion, assume that an arbitrary graph $\mathcal{G} = (V, E)$, not necessarily triangulated, is given. Then, for any vertex $v \in V$, *eliminating* v from the graph $\mathcal{G}$ involves two steps: (1) add edges to $\mathcal{G}$ so that $N_{\mathcal{G}}(v)$ becomes a clique; (2) remove vertex v and all incident edges from this new graph. The first step can be accomplished by adding the edges contained in $D_{\mathcal{G}}(v)$; the second step generates the subgraph induced by the vertices $V - \{v\}$. More formally, these steps may be written as follows,

$$\mathcal{G}' = (V, E \cup D_{\mathcal{G}}(v)) \tag{3.26a}$$

$$\mathcal{G}^{\downarrow} = \mathcal{G}'(V - \{v\}), \tag{3.26b}$$

where $\mathcal{G}'$ is an intermediate graph generated in the process of elimination and $\mathcal{G}^{\downarrow}$ is the resulting *elimination graph*. We henceforth use the notation $\downarrow(\mathcal{G}, v)$ to denote the elimination graph obtained by eliminating vertex v from $\mathcal{G}$. Figure 3.8 provides a graphical illustration of the steps involved in vertex elimination.

Extending this idea, we now consider the process of eliminating a subset of the vertices of a graph $\mathcal{G} = (V, E)$. Specifically, given a set $A \subset V$, the vertices in $V - A$ are eliminated one-by-one, and the resulting subgraph is denoted by $\mathcal{G}\langle A\rangle$. It can be shown that the graph $\mathcal{G}\langle A\rangle$ is independent of the order in which the vertices $V - A$ are eliminated [82]. An illustration of this invariance property is shown in Figures 3.9 and 3.10, where the elimination graphs in Figures 3.9(c) and 3.10(c) are identical yet obtained by eliminating vertices 1 and 2 in a different order.

Note that the subgraph induced by the set A, *i.e.* $\mathcal{G}(A)$, is not necessarily the same as $\mathcal{G}\langle A\rangle$ since additional edges may be added in the process of vertex elimination. In addition, notice that $\mathcal{G}\langle V - \{v\}\rangle$ and $\downarrow(\mathcal{G}, v)$ are equivalent notations for the elimination graph obtained by eliminating vertex v.

Having defined vertex elimination, we are now in a position to discuss the elimination game, which simply involves a sequence of vertex eliminations. To define such a sequence, an *ordering* on the vertices is required: an ordering α on a set of vertices V is a bijection of the form $\alpha : \{1, \ldots, n\} \to V$, where $n = |V|$. Given an ordering α, we define a sequence of elimination graphs as follows

$$\mathcal{G}_0^{\downarrow} \triangleq \mathcal{G} \tag{3.27a}$$

$$\mathcal{G}_i^{\downarrow} \triangleq \downarrow\left(\mathcal{G}_{i-1}^{\downarrow}, \alpha(i)\right) = \mathcal{G}\langle V - \{\alpha(1), \ldots, \alpha(i)\}\rangle, \quad i = 1, \ldots, n. \tag{3.27b}$$

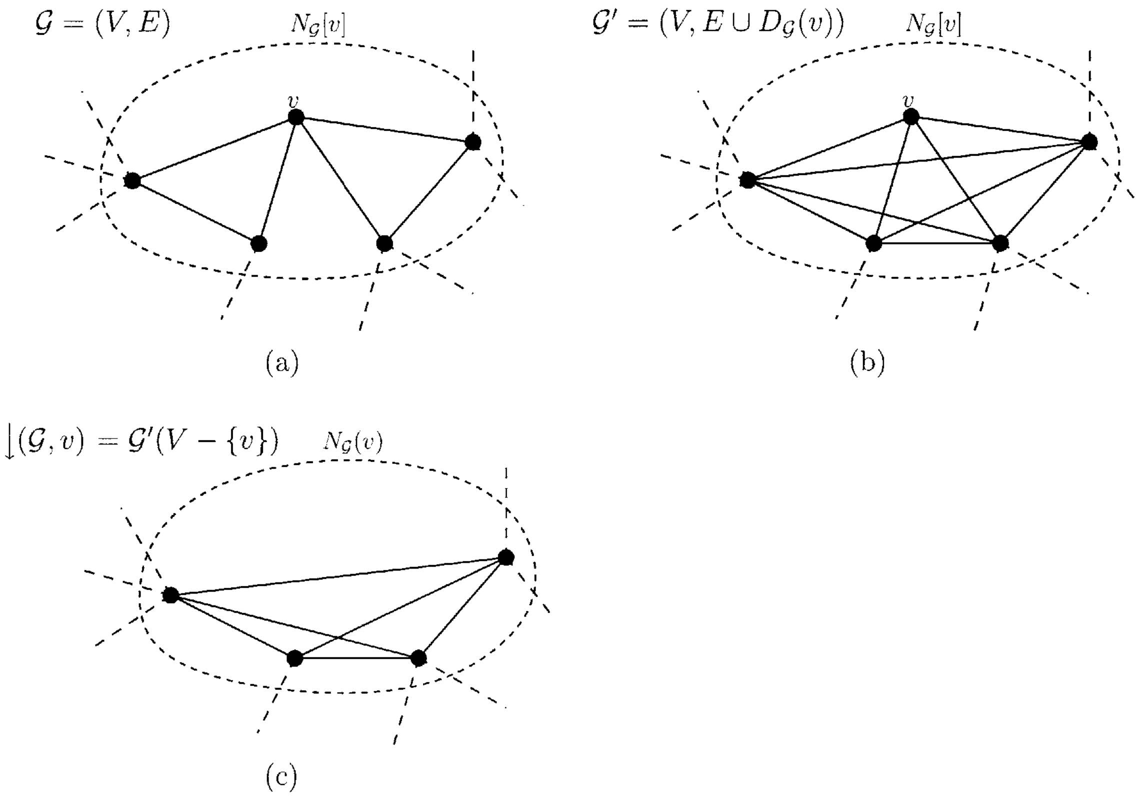

$\mathcal{G} = (V, E)$
$N_{\mathcal{G}}[v]$
v
(a)
$\mathcal{G}' = (V, E \cup D_{\mathcal{G}}(v))$
$N_{\mathcal{G}}[v]$
v
(b)
$\downarrow(\mathcal{G}, v) = \mathcal{G}'(V - \{v\})$
$N_{\mathcal{G}}(v)$
(c)

Figure 3.8. Illustration of the steps involved in vertex elimination. (a) A graph $\mathcal{G} = (V, E)$ is given. (b) Graph $\mathcal{G}'$ is formed by adding edges such that $N_{\mathcal{G}}(v)$ becomes a clique. (c) Vertex v and all incident edges are removed from the graph to give the elimination graph $\downarrow(\mathcal{G}, v)$.

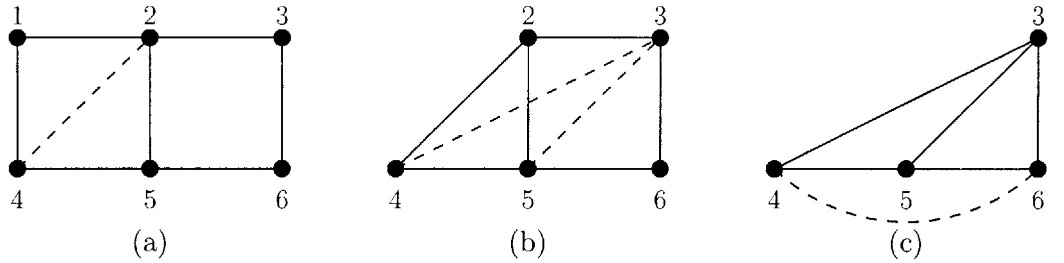


Figure 3.9. Graphical illustration of a sequence of elimination graphs for the vertex ordering $\alpha = (1, 2, 3, 4, 5, 6)$. The dashed edges indicate the elimination deficiencies. (a) $\mathcal{G}_0^{\downarrow} = \mathcal{G}$ (solid), $D_{\mathcal{G}}^{\downarrow}(1)$ (dashed) (b) $\mathcal{G}_1^{\downarrow}$ (solid), $D_{\mathcal{G}}^{\downarrow}(2)$ (dashed) (c) $\mathcal{G}_2^{\downarrow}$ (solid), $D_{\mathcal{G}}^{\downarrow}(3)$ (dashed)

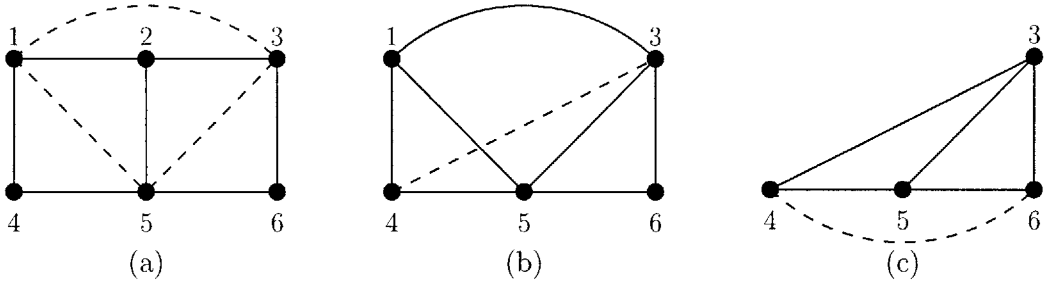


Figure 3.10. Graphical illustration of a sequence of elimination graphs for the vertex ordering $\alpha = (2, 1, 3, 4, 5, 6)$. The dashed edges indicate the elimination deficiencies. (a) $\mathcal{G}_0^{\downarrow} = \mathcal{G}$ (solid), $D_{\mathcal{G}}^{\downarrow}(2)$ (dashed) (b) $\mathcal{G}_1^{\downarrow}$ (solid), $D_{\mathcal{G}}^{\downarrow}(1)$ (dashed) (c) $\mathcal{G}_2^{\downarrow}$ (solid), $D_{\mathcal{G}}^{\downarrow}(3)$ (dashed)

Therefore, the graph $\mathcal{G}_i^{\downarrow}$ is obtained from $\mathcal{G}$ by eliminating the vertices $\{\alpha(1), \dots, \alpha(i)\}$, or equivalently, $\mathcal{G}_i^{\downarrow}$ is obtained from $\mathcal{G}_{i-1}^{\downarrow}$ by eliminating vertex $\alpha(i)$. Even though the sequence (3.27) is a function of α, we suppress this dependence for notational clarity; the particular choice of α will be clear from context. Figures 3.9 and 3.10 illustrate two sequences of elimination graphs for different vertex orderings.

For our purposes, we are less interested in the sequence of graphs $\{\mathcal{G}_i^{\downarrow}\}$ than in the edges added in the process of elimination. Specifically, consider the two graphs $\mathcal{G}_{i-1}^{\downarrow}$ and $\mathcal{G}_i^{\downarrow}$. To eliminate $\alpha(i)$ from $\mathcal{G}_{i-1}^{\downarrow}$, the edges in $D_{\mathcal{G}_{i-1}^{\downarrow}}(\alpha(i))$ must be added to $\mathcal{G}_{i-1}^{\downarrow}$in order to make $N_{\mathcal{G}_{i-1}^{\downarrow}}(\alpha(i))$ a clique. We call these added edges *fill*, and we introduce notation to denote this fill. In particular, given an ordering α, the *elimination deficiency* of vertex $v = \alpha(i)$ is defined as

$$D_{\mathcal{G}}^{\downarrow}(v) \triangleq D_{\mathcal{G}_{i-1}^{\downarrow}}(v), \quad v = \alpha(i), \tag{3.28}$$

i.e. it is the deficiency of vertex $\alpha(i)$ in the elimination graph $\mathcal{G}_{i-1}^{\downarrow}$. Similarly, the *elimination neighborhood* of vertex $v = \alpha(i)$ is defined as

$$N_{\mathcal{G}}^{\downarrow}(v) \triangleq N_{\mathcal{G}_{i-1}^{\downarrow}}(v), \quad v = \alpha(i), \tag{3.29a}$$

$$N_{\mathcal{G}}^{\downarrow}[v] \triangleq N_{\mathcal{G}}^{\downarrow}(v) \cup \{v\}. \tag{3.29b}$$

Even though both $D_{\mathcal{G}}^{\downarrow}(v)$ and $N_{\mathcal{G}}^{\downarrow}(v)$ are functions of α, we again choose to suppress this dependence for clarity.

As an illustration, the dashed lines in Figures 3.9(a),(b), and (c) indicate the edges contained in $D_{\mathcal{G}}^{\downarrow}(1)$, $D_{\mathcal{G}}^{\downarrow}(2)$, and $D_{\mathcal{G}}^{\downarrow}(3)$ respectively, for the ordering $\alpha = (1, 2, 3, 4, 5, 6)$.[20] The dashed lines in Figures 3.10(a),(b), and (c) indicate the edges contained in $D_{\mathcal{G}}^{\downarrow}(2)$, $D_{\mathcal{G}}^{\downarrow}(1)$, and $D_{\mathcal{G}}^{\downarrow}(3)$ respectively, for the ordering $\alpha = (2, 1, 3, 4, 5, 6)$. Notice that while the elimination graph $\mathcal{G}_2^{\downarrow}$ is identical in both figures, the edges added during the elimination process are somewhat different.

In general, the number of edges added during the elimination process is dependent on the initial graph $\mathcal{G}$ and the ordering α. For example, as we discuss in the next section, if the initial

[20] Recall from the previous chapter that we used the ordered set $(v_1, \dots, v_m)$ to denote a particular ordering on the non-leaf vertices. We also use this notation here, when it is necessary for us to state a specific ordering α on the full set of vertices.

graph $\mathcal{G}$ is triangulated then there exists an ordering α such that no fill edges are generated during the elimination game. Also, if the initial graph $\mathcal{G}$ is not triangulated, then adding all of the fill edges to $\mathcal{G}$ will generate a triangulated graph. Specifically, given a graph $\mathcal{G} = (V, E)$ (not necessarily triangulated) and an ordering α, if we define $F \triangleq \cup_{v \in V} D^{\downarrow}_{\mathcal{G}}(v)$, then $\mathcal{G}' = (V, E \cup F)$ is triangulated [82]. Triangulated graphs and vertex orderings are further examined in the next section.

■ 3.6.2 Elimination Orderings

As mentioned above, the degree of fill generated during the elimination game is closely tied to the original graph $\mathcal{G}$ and the ordering α on the vertices. If there exists an ordering α such that no fill is generated during the elimination process, then we call α a *perfect elimination ordering.* More specifically, α is a perfect elimination ordering for a graph $\mathcal{G} = (V, E)$ if $D^{\downarrow}_{\mathcal{G}}(v) = \{\emptyset\}$ for all $v \in V$. The class of graphs $\mathcal{G}$ for which a perfect elimination ordering always exists is precisely the class of triangulated graphs, as evidenced by the following well-known result.

Theorem 3.5 (Triangulated Graphs and Perfect Elimination Orderings).
A graph is triangulated if and only if there exists a perfect elimination ordering on the vertices.

Proof. See [89]. ■

For instance, the graph represented by the solid lines in Figure 3.9(a) is not triangulated. In fact, no vertex in the graph is simplicial, implying that no vertex can even start a perfect elimination ordering. If the dashed edges shown in Figures 3.9(a)–(c) are added to the graph in Figure 3.9(a), then the resulting graph is triangulated, since by construction $\alpha = (1, 2, 3, 4, 5, 6)$ is a perfect elimination ordering.

In subsequent sections, we occasionally place additional restrictions on an ordering α. For example, given a graph $\mathcal{G} = (V, E)$ with $|V| = n$ and a set $A \subset V$ with $|A| = n - k$, we sometimes require the first k vertices of α, *i.e.* $\alpha(1), \ldots, \alpha(k)$, to provide an ordering on the set $V - A$, while the last $n - k$ elements of α, *i.e.* $\alpha(k + 1), \ldots, \alpha(n)$, to provide an ordering on the set A. If this is the case, we say that α is an *elimination ordering down to A*, *i.e.* $\alpha(j) \in A$ for $j = k + 1, \ldots, n$. If in addition α is a perfect elimination ordering, we say that α is a *perfect elimination ordering down to A*. The following result indicates that for triangulated graphs we can always find a perfect elimination ordering down to A if A corresponds to a clique of the graph.

Lemma 3.4 (Perfect Elimination Down to a Clique).
Let $\mathcal{G} = (V, E)$ be a triangulated graph with clique $C \subseteq V$. There exists a perfect elimination ordering down to C.

Proof. See [89]. ■

In some instances, we are not interested in the properties of a complete ordering of the vertices but rather the first k vertices in an ordering. We say that α is a *k-partial elimination ordering* if $D^{\downarrow}_{\mathcal{G}}(\alpha(i)) = \{\emptyset\}$ for $i = 1, \ldots, k$. In other words, $\alpha(1), \ldots, \alpha(k)$ is the start of a perfect elimination ordering. Of course, a perfect elimination ordering is always a k-partial elimination ordering, but a k-partial elimination ordering may or may not be a perfect elimination ordering. When the value of k is not important, we refer to a k-partial elimination ordering as simply a *partial elimination ordering.*

Given a graph $\mathcal{G} = (V, E)$ and a set $A \subset V$, we stated earlier that the induced subgraph $\mathcal{G}(A)$ and the elimination graph $\mathcal{G}\langle A\rangle$ are not necessarily the same. Using the preceding definitions, we can state conditions under which the two graphs are identical.

Lemma 3.5 (Induced Subgraphs and Elimination Graphs).
Given $\mathcal{G} = (V, E)$ and a set $A \subset V$ with $|V| - k = |A|$, the two graphs $\mathcal{G}(A)$ and $\mathcal{G}\langle A\rangle$ are identical if there exists a k-partial elimination ordering α down to A.

Proof. If α is a k-partial elimination ordering, no edges are added in the process of elimination down to A, and therefore, $\mathcal{G}\langle A\rangle$ is by definition the subgraph induced by A. ■

■ 3.6.3 Vertex Elimination and Marginalization

This section relates the graph-theoretic notion of vertex elimination to the probabilistic notion of marginalization. For our purposes, it is sufficient to examine this relationship for the class of triangulated graphs and for the special case where the vertex to be eliminated is simplicial. The fact that the graphs are triangulated allows us to obtain an explicit factorization of the densities in terms of their marginals. In addition, since we consider simplicial vertices, the process of vertex elimination does not introduce additional edges in the elimination graph, and as we show here, the resulting marginal density does not possess additional dependencies beyond those contained in the original density.

A Factorization Involving Simplicial Vertices

Let $\mathcal{G} = (V, E)$ be an arbitrary triangulated graph, and let $p(x_V)$ be a probability density indexed by V. As discussed earlier, $p_\mathcal{G}$ is a well-defined probability density, and as such, we can write

$$p_\mathcal{G}(x) = p_\mathcal{G}(x_v | x_{V-\{v\}}) p_\mathcal{G}(x_{V-\{v\}}). \tag{3.30}$$

In addition, $p_\mathcal{G}$ by definition respects the Markov properties implied by the graph $\mathcal{G}$. Consequently, the separation properties of $\mathcal{G}$ may be used to characterize the density $p_\mathcal{G}(x_v | x_{V-\{v\}})$, and doing so, gives the following

$$p_\mathcal{G}(x_v | x_{V-\{v\}}) = p_\mathcal{G}(x_v | x_{N_\mathcal{G}(v)}). \tag{3.31}$$

This indicates that $X_{N_\mathcal{G}(v)}$ is a sufficient statistic for $X_{V-\{v\}}$ (under the density $p_\mathcal{G}$) and follows from the fact that $N_\mathcal{G}(v)$ separates vertex v from the rest of the vertices in $\mathcal{G}$. Combining (3.30) and (3.31) provides a functional relationship between the full density $p_\mathcal{G}(x)$ and the marginal density $p_\mathcal{G}(x_{V-\{v\}})$.

The important consequence of (3.31) is the fact that $p_\mathcal{G}(x_v | x_{V-\{v\}})$ is only a function of the "local" variables X_v and $X_{N_\mathcal{G}(v)}$. However, a closed-form expression for this function (in terms of the original density p) is not generally possible because of the fact that $p_\mathcal{G}(x_v | x_{N_\mathcal{G}(v)})$ is obtained by marginalizing the density $p_\mathcal{G}$. If vertex v is simplicial, though, this integration is not necessary, and we can immediately write $p_\mathcal{G}(x_v | x_{V-\{v\}}) = p(x_v | x_{N_\mathcal{G}(v)})$, a fact which we now prove.

Let $\mathcal{T} = (\mathcal{C}, \mathcal{S})$ be the junction tree representation for $\mathcal{G} = (V, E)$, and let vertex $v \in V$ be simplicial in $\mathcal{G}$. Since v is simplicial, Lemma 3.1 indicates that $N_\mathcal{G}[v]$ is the unique maximal clique of $\mathcal{G}$ containing v. If we define $C \triangleq N_{\mathcal{G}}[v]$, then the following decomposition for $p_{\mathcal{G}}(x)$ holds,

$$\begin{aligned} p_{\mathcal{G}}(x) &= \frac{\prod_{C' \in \mathcal{C}} p(x_{C'})}{\prod_{S' \in \mathcal{S}} p(x_{S'})} = p(x_C) \frac{\prod_{C' \in \mathcal{C}-\{C\}} p(x_{C'})}{\prod_{S' \in \mathcal{S}} p(x_{S'})} \\ &= p(x_v | x_{N_{\mathcal{G}}(v)}) \left[p(x_{N_{\mathcal{G}}(v)}) \frac{\prod_{C' \in \mathcal{C}-\{C\}} p(x_{C'})}{\prod_{S' \in \mathcal{S}} p(x_{S'})} \right]. \end{aligned} \tag{3.32}$$

Since v is part of the unique maximal clique C, the bracketed expression in (3.32) does not contain the variable x_v. Consequently, integrating out x_v from both (3.30) and (3.32) shows that $p_{\mathcal{G}}(x_{V-\{v\}})$ is equal to the term in brackets. Thus, when vertex v is simplicial in $\mathcal{G}$, the following probabilistic relationship holds,

$$p_{\mathcal{G}}(x) = p(x_v | x_{N_{\mathcal{G}}(v)}) p_{\mathcal{G}}(x_{V-\{v\}}). \tag{3.33}$$

Simplicial Vertices and Marginalization

Another interesting thing happens when v is simplicial. Specifically, we are able to easily characterize the independence structure of the marginal density $p_{\mathcal{G}}(x_{V-\{v\}})$. Namely, we can write $p_{\mathcal{G}}(x_{V-\{v\}}) = p_{\mathcal{G}^{\downarrow}}(x)$ where $\mathcal{G}^{\downarrow} = \downarrow(\mathcal{G}, v)$,[21] and therefore, the graph $\mathcal{G}^{\downarrow}$ characterizes all of the independencies present in the marginal density $p_{\mathcal{G}}(x_{V-\{v\}})$. To show this, we first examine the junction tree representation, and we prove that it changes in a predictable fashion when a simplicial vertex $v \in V$ is eliminated from a triangulated graph. In particular, the following proposition indicates that the junction tree representation can change in only one of two ways.

Proposition 3.5 (Vertex Elimination and Junction Trees).
Let $\mathcal{G} = (V, E)$ be a triangulated graph, and let $\mathcal{T} = (\mathcal{C}, \mathcal{S})$ be the junction tree representation of $\mathcal{G}$. Suppose $v \in V$ is a simplicial vertex, and let C denote the unique maximal clique containing v. If we define the elimination graph $\mathcal{G}^{\downarrow} \triangleq \downarrow(\mathcal{G}, v)$ as well as the sets $C^{\downarrow} \triangleq C - \{v\}$ and $\mathcal{C}^{\downarrow} \triangleq \mathcal{C} - \{C\}$, then one and only one of the following is a junction tree representation $\mathcal{T}^{\downarrow}$ for $\mathcal{G}^{\downarrow}$:

(1) $\mathcal{T}^{\downarrow} = (\mathcal{C}^{\downarrow} \cup \{C^{\downarrow}\}, \mathcal{S})$,

(2) $\mathcal{T}^{\downarrow} = (\mathcal{C}^{\downarrow}, \mathcal{S} - \{C^{\downarrow}\})$.

Proof. See Appendix B.1. ■

The first junction tree representation occurs when $C - \{v\}$ is a maximal clique in the elimination graph $\mathcal{G}^{\downarrow}$, while the second occurs when $C - \{v\}$ is a subset of another maximal clique in $\mathcal{G}^{\downarrow}$.

To illustrate these two possibilities, consider the graphs and junction trees shown in Figure 3.11. The original triangulated graph $\mathcal{G}$ and a corresponding junction tree are shown in Figures 3.11(a) and (b) respectively. Notice that $\{1, 2, 4, 5\}$ is a maximal clique in $\mathcal{G}$. If vertex 1 is eliminated from $\mathcal{G}$ to form $\mathcal{G}^{\downarrow}$, then $\{2, 4, 5\}$ becomes a maximal clique of $\mathcal{G}^{\downarrow}$ as illustrated in Figure 3.11(c), and because of this, the junction tree simply reflects this change in the set of maximal cliques, as illustrated in Figure 3.11(d). Consider now what happens when vertex 6 is eliminated from $\mathcal{G}$ as shown in Figure 3.11(e). In this case, $\{3, 5, 6\}$ is a maximal clique of $\mathcal{G}$, but when vertex 6 is eliminated to form $\mathcal{G}^{\downarrow}$, $\{3, 5\}$ is a subset of the maximal clique $\{2, 3, 5\}$ in $\mathcal{G}^{\downarrow}$. Consequently, the

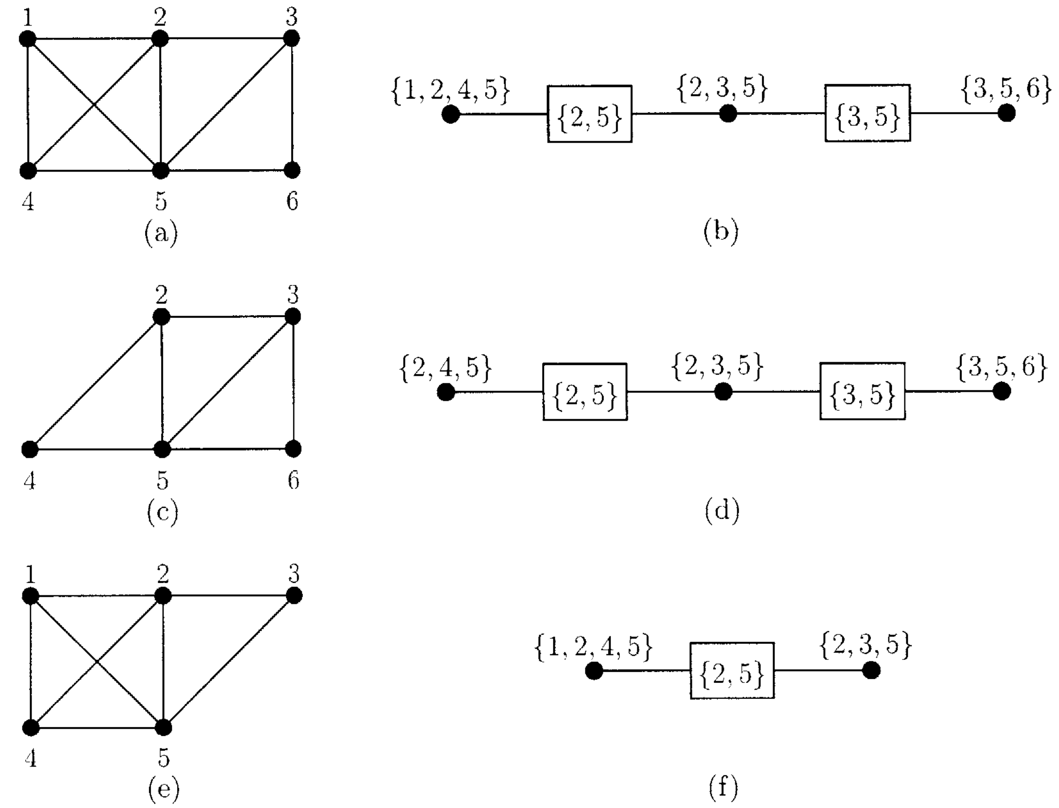


Figure 3.11. Graphical illustration of the two possible junction trees which can result from removing a simplicial vertex from a triangulated graph. (a) The original graph $\mathcal{G}$. (b) A junction tree for the graph $\mathcal{G}$ in (a). (c) The elimination graph $\mathcal{G}^{\downarrow} = \downarrow(\mathcal{G}, 1)$. (d) A junction tree for $\mathcal{G}^{\downarrow}$ in (c). The maximal clique $\{1, 2, 4, 5\}$ in $\mathcal{G}$ is replaced by the new maximal clique $\{2, 4, 5\}$. (e) The elimination graph $\mathcal{G}^{\downarrow} = \downarrow(\mathcal{G}, 6)$. (f) A junction tree for $\mathcal{G}^{\downarrow}$ in (e). The maximal clique $\{3, 5, 6\}$ as well as the separator $\{3, 5\}$ have been eliminated.

junction tree no longer has the maximal clique $\{3,5,6\}$ as well as the separator $\{3,5\}$ as shown in Figure 3.11(f).

Returning to our earlier goal, we now show that the marginal density $p_{\mathcal{G}}(x_{V-\{v\}})$ (or equivalently the bracketed expression in (3.32)) is equal to $p_{\mathcal{G}^\downarrow}(x)$ with $\mathcal{G}^\downarrow = \downarrow(\mathcal{G}, v)$. Consider the two cases for the junction tree representation of $\mathcal{G}^\downarrow$ as indicated in Proposition 3.5. In the first case, the separator sets for $\mathcal{G}^\downarrow$ are the same as the separators $\mathcal{S}$ of $\mathcal{G}$, and the set of maximal cliques of $\mathcal{G}^\downarrow$ are given by $(\mathcal{C} - \{C\}) \cup \{N_{\mathcal{G}}(v)\}$, where we have used the fact that $C - \{v\} = N_{\mathcal{G}}(v)$. Consequently, the bracketed expression in (3.32) is by definition equal to $p_{\mathcal{G}^\downarrow}(x)$. In the second case provided in Proposition 3.5, the set of maximal cliques of $\mathcal{G}^\downarrow$ is given by $\mathcal{C} - \{C\}$, and the set of separators is given by $\mathcal{S} - \{N_{\mathcal{G}}(v)\}$. We also know from the proof to Proposition 3.5 that, in this second case, at least one of the separator sets of $\mathcal{G}$ is equal to $N_{\mathcal{G}}(v)$. Consequently, the bracketed expression in (3.32) is by definition equal to $p_{\mathcal{G}^\downarrow}(x)$ because $p(x_{N_{\mathcal{G}}(v)})$ cancels with one of the terms $p(x_S)$.

Hence, we have proven the following important relationship,

$$p_{\mathcal{G}}(x) = p(x_v|x_{N_{\mathcal{G}}(v)})p_{\mathcal{G}^\downarrow}(x). \tag{3.34}$$

Since v is simplicial, we also know that the elimination graph $\mathcal{G}^\downarrow$ is equal to the subgraph $\mathcal{G}(V-\{v\})$, and consequently, since $p_{\mathcal{G}}(x_{V-\{v\}}) = p_{\mathcal{G}^\downarrow}(x)$, the marginal density has no additional dependencies than those exhibited by the original density $p_{\mathcal{G}}$. All of these ideas are summarized in the following proposition.

Proposition 3.6 (Vertex Elimination and Marginalization).
Let $\mathcal{G} = (V, E)$ be a triangulated graph. For $v \in V$, define the elimination graph $\mathcal{G}^\downarrow \triangleq \downarrow(\mathcal{G}, v)$. Then, the following relationship holds

$$p_{\mathcal{G}}(x) = p_{\mathcal{G}}(x_v|x_{N_{\mathcal{G}}(v)})p_{\mathcal{G}}(x_{V-\{v\}}), \tag{3.35}$$

and if v is a simplicial vertex in $\mathcal{G}$, then

$$p_{\mathcal{G}}(x) = p(x_v|x_{N_{\mathcal{G}}(v)})p_{\mathcal{G}}(x_{V-\{v\}}) = p(x_v|x_{N_{\mathcal{G}}(v)})p_{\mathcal{G}^\downarrow}(x). \tag{3.36}$$

Proof. See the preceding discussion. ∎

The relationship in (3.36) may also be applied in a recursive fashion to a sequence of elimination graphs, as long as a simplicial vertex is removed at each step. The following corollary states this extension of Proposition 3.6 for a k-partial elimination ordering α.

Corollary 3.1 (k-Partial Elimination Orderings and Marginalization).
Let $\mathcal{G} = (V, E)$ be a triangulated graph. Suppose α is a k-partial elimination ordering, and define the elimination graph $\mathcal{G}^\downarrow \triangleq \mathcal{G}\left\langle V - \{\alpha(1), \ldots, \alpha(k)\}\right\rangle$. Then, the following decomposition holds for $p_{\mathcal{G}}(x)$,

$$p_{\mathcal{G}}(x) = p_{\mathcal{G}^\downarrow}(x)\prod_{i=1}^{k} p\left(x_{\alpha(i)}|x_{N^\downarrow_{\mathcal{G}}(\alpha(i))}\right) = p_{\mathcal{G}}\left(x_{V-\{\alpha(1),\ldots,\alpha(k)\}}\right)\prod_{i=1}^{k} p\left(x_{\alpha(i)}|x_{N^\downarrow_{\mathcal{G}}(\alpha(i))}\right).$$

Proof. Follows directly by recursively applying (3.36). See Appendix B.2 for details. ∎

[21] Recall from the discussion in Section 3.3.2 that the shorthand $p_{\mathcal{G}^\downarrow}(x)$ is often used instead of $p_{\mathcal{G}^\downarrow}(x_{V-\{v\}})$.

3.7 Clique Extensions and Neighborhood Separators

3.7.1 Clique Extensions

The relationship between vertex elimination and marginalization, established in the preceding section, may be used to derive a simple functional relationship between two densities $p_{\mathcal{G}}$ and $p_{\mathcal{G}'}$ for some choices of $\mathcal{G}$ and $\mathcal{G}'$. In this section, we focus on triangulated graphs $\mathcal{G}$ and $\mathcal{G}'$ where $\mathcal{G}'$ is a supergraph of $\mathcal{G}$. The fact that $\mathcal{G}'$ has the same edges as $\mathcal{G}$ plus additional edges implies that $p_{\mathcal{G}'}$ exhibits fewer conditional independencies than $p_{\mathcal{G}}$, and as we show, these conditional independencies are easy to characterize due to the structure of the graphs $\mathcal{G}'$ considered here. In particular, the relationship between $p_{\mathcal{G}}$ and $p_{\mathcal{G}'}$ is simple because $\mathcal{G}'$ is not an arbitrary supergraph of $\mathcal{G}$ but is contained within a subclass of all supergraphs called *clique extensions*.

Definition 3.3 (Clique Extensions).
Let $\mathcal{G} = (V, E)$ be an arbitrary triangulated graph. A graph $\mathcal{G}' = (V, E')$ is called a *clique extension* of $\mathcal{G}$ if $\mathcal{G}'$ is a triangulated supergraph of $\mathcal{G}$ with only one additional maximal clique not contained in $\mathcal{G}$. ◀

Note that any supergraph $\mathcal{G}'$ of a graph $\mathcal{G}$ must have at least one new maximal clique not contained in $\mathcal{G}$. The key aspect of a clique extension is that it only has one additional maximal clique. Of course, this maximal clique might be formed by joining together many smaller cliques into a new larger clique, and consequently, the junction tree representations for $\mathcal{G}$ and $\mathcal{G}'$ can be significantly different. As we later show, however, it is not necessary to explicitly examine the junction tree representation in order to derive a probabilistic relationship between $p_{\mathcal{G}}$ and $p_{\mathcal{G}'}$.

One Type of Clique Extension

For a general triangulated graph $\mathcal{G}$, it may be difficult to characterize which edges can be added to $\mathcal{G}$ in order to generate a clique extension. As it turns out, we have already discussed one type of clique extension. Namely, if we choose some vertex $v \in V$ and add all of the edges in $D_{\mathcal{G}}(v)$ to the graph $\mathcal{G}$, then the resulting graph is a clique extension with new maximal clique $N_{\mathcal{G}}[v]$. For example, consider the graph $\mathcal{G}$ shown in Figure 3.12(a), where the solid lines correspond to edges of $\mathcal{G}$. If the dashed edges corresponding to $D_{\mathcal{G}}(2)$ are added to the graph, then the resulting graph is a clique extension with new maximal clique $N_{\mathcal{G}}[2] = \{1, 2, 3, 5, 6\}$.

In subsequent sections, we consider clique extensions like that generated by adding edges $D_{\mathcal{G}}(2)$ to the graph in Figure 3.12(a); however, we may not add all of the edges in the set $D_{\mathcal{G}}(2)$. Instead, we may choose to add only a subset of these edges. Consider the three possibilities which could result from choosing such a subset, as illustrated in Figures 3.12(b)–(d). If only edge $\{1, 3\}$ is added to the graph, then the resulting graph is not triangulated because the cycle $[1, 3, 6, 5, 1]$ has no chord. If edges $\{1, 3\}$ and $\{3, 5\}$ are added, the resulting graph is triangulated but not a clique extension since two new maximal cliques $\{2, 3, 5, 6\}$ and $\{1, 2, 3, 5\}$ are generated. Finally, if edge $\{1, 6\}$ is added, the resulting graph is a clique extension with the unique new maximal clique $\{1, 2, 5, 6\}$.

Given that choosing only a subset of the edges in $D_{\mathcal{G}}(2)$ can generate three such possibilities, we now formally characterize when such a choice will generate a clique extension; this characterization is provided in Proposition 3.7. Before that, though, we graphically illustrate this characterization. Consider the graph $\mathcal{G}$ in Figure 3.12(a), and notice that vertices 3 and 4 are the start of a perfect elimination ordering for $\mathcal{G}$. If we define $\mathcal{G}^{\downarrow} \triangleq \mathcal{G}\left\langle\{1, 2, 5, 6\}\right\rangle = \mathcal{G}\left(\{1, 2, 5, 6\}\right)$, then the deficiency

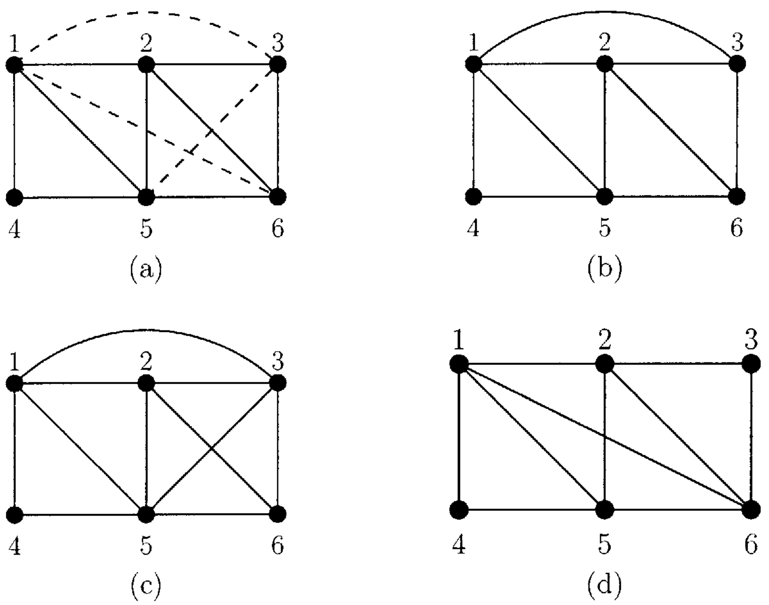


Figure 3.12. (a) The solid lines correspond to a triangulated graph $\mathcal{G}$, while the dashed lines correspond to edges contained in $D_{\mathcal{G}}(2)$. If all of the dashed edges are added to the graph, the resulting graph is a clique extension with new maximal clique $\{1,2,3,5,6\}$. If only a subset of the edges in $D_{\mathcal{G}}(2)$ are added to $\mathcal{G}$ then the resulting graph has different properties depending on which edges are chosen: (b) Edge $\{1,3\}$ added; the graph is not triangulated. (c) Edges $\{1,3\}$ and $\{3,5\}$ added; the graph is triangulated but not a clique extension since two new maximal cliques $\{2,3,5,6\}$ and $\{1,2,3,5\}$ are formed. (d) Edge $\{1,6\}$ added; the graph is a clique extension with new maximal clique $\{1,2,5,6\}$.

of vertex 2 in this new graph is given by $D_{\mathcal{G}^\downarrow}(2) = \{\{1,6\}\}$, *i.e.* $\mathcal{G}^\downarrow$ is a subgraph of $\mathcal{G}$ where the deficiency of vertex 2 is the single edge $\{1,6\}$. Adding the edge $\{1,6\}$ to the graph generates the clique extension shown in Figure 3.12(d).

More generally, Proposition 3.7 shows that a set of edges $F \subseteq D_{\mathcal{G}}(v)$ added to a graph $\mathcal{G}$ will generate a clique extension if and only if there exists an elimination graph of $\mathcal{G}$ (obtained by successively eliminating simplicial vertices) where F is equal to the deficiency of v in the elimination graph. Notice that the graphs in Figures 3.12(b) and (c) are not clique extensions of $\mathcal{G}$ because no such elimination graph exists.

Proposition 3.7 (Elimination Graphs and Clique Extensions).
Let $\mathcal{G} = (V, E)$ be a triangulated graph. For some $v \in V$, let $F \subseteq D_{\mathcal{G}}(v)$, $F \neq \{\emptyset\}$, and define the new graph $\mathcal{G}' \triangleq (V, E \cup F)$. Then, $\mathcal{G}'$ is a clique extension of $\mathcal{G}$ if and only if there exists a k-partial elimination ordering α of $\mathcal{G}$ such that $F = D_{\mathcal{G}^\downarrow}(v)$, with $\mathcal{G}^\downarrow \triangleq \mathcal{G}\langle V - \{\alpha(1), \ldots, \alpha(k)\}\rangle$. Furthermore, the unique new maximal clique C contained in $\mathcal{G}'$ is given by $C = N_{\mathcal{G}^\downarrow}[v]$.

Proof. See Appendix B.3. ■

Graph Structure of Clique Extensions

Having defined clique extensions, we now provide several important properties of clique extensions. The first property involves what we call the *internal structure* of a clique extension, namely the structure within the new maximal clique. The following proposition states that every edge $\{a, b\}$ contained in the clique extension $\mathcal{G}'$ but not $\mathcal{G}$ must satisfy $\{a, b\} \subset C$, where C is the new maximal clique of $\mathcal{G}'$. Simply stated, this constraint must be satisfied because otherwise more than one maximal clique would be formed.

Proposition 3.8 (Internal Structure of Clique Extensions).
Let $\mathcal{G} = (V, E)$ be a triangulated graph, and let $\mathcal{G}' = (V, E \cup F)$ with $E \cap F = \emptyset$ be a clique extension of $\mathcal{G}$. Suppose C is the unique maximal clique contained in $\mathcal{G}'$ but not $\mathcal{G}$. If $\{a, b\} \in F$ then $\{a, b\} \subset C$.

Proof. Suppose $\{a, b\} \in F$ and $\{a, b\} \not\subset C$, then $\{a, b\} \subset C'$, where $C' \neq C$ is a maximal clique of $\mathcal{G}'$, and since $\{a, b\} \notin E$, C' is not a maximal clique of $\mathcal{G}$. This contradicts the fact that $\mathcal{G}'$ is a clique extension of $\mathcal{G}$. ■

Hence, this first property indicates that the edges of the two graphs $\mathcal{G}$ and $\mathcal{G}'$ differ only within the maximal clique C.

We now consider the *external structure* of a clique extension, *i.e.* the structure outside the maximal clique C. The following proposition provides two related properties of external structure.

Proposition 3.9 (External Structure of Clique Extensions).
Let $\mathcal{G}$, $\mathcal{G}'$, and C be defined as in Proposition 3.8. Let α be a perfect elimination ordering down to C for the graph $\mathcal{G}'$. If $k = |V| - |C|$, then:

(1) $N^{\downarrow}_{\mathcal{G}}(\alpha(i)) = N^{\downarrow}_{\mathcal{G}'}(\alpha(i))$ for $i = 1, \ldots, k$,

(2) α is also a k-partial elimination ordering for $\mathcal{G}$.

Proof.

(1) By Lemma 3.4, we know that a perfect elimination ordering α down to C exists for the graph $\mathcal{G}'$. If $N^{\downarrow}_{\mathcal{G}}(\alpha(i)) \neq N^{\downarrow}_{\mathcal{G}'}(\alpha(i))$ for some $i = 1, \ldots, k$, then there exists an edge $\{\alpha(i), b\} \in E \cup F$ and $\{\alpha(i), b\} \notin E$. Hence, $\{\alpha(i), b\} \in F$, and by Proposition 3.8, we must have $\{\alpha(i), b\} \subset C$, which contradicts the fact that $\alpha(i) \notin C$ for $i = 1, \ldots, k$.

(2) Suppose α is not a k-partial elimination ordering for $\mathcal{G}$. Then, $D^{\downarrow}_{\mathcal{G}}(\alpha(i)) \neq \{\emptyset\}$ for at least one $i = 1, \ldots, k$. Thus, we have at least one edge $\{a, b\} \in D^{\downarrow}_{\mathcal{G}}(\alpha(i))$ and $\{a, b\} \notin D^{\downarrow}_{\mathcal{G}'}(\alpha(i))$, and consequently, $\{a, b\}$ is not an edge in $\mathcal{G}$ but is an edge in $\mathcal{G}'$. This implies that $\{\alpha(i), a, b\}$ is a clique in $\mathcal{G}'$ (but not $\mathcal{G}$) and therefore a subset of some maximal clique; call this maximal clique C'. Now, we have $\alpha(i) \in C'$ and $\alpha(i) \notin C$ (by the definition of the ordering α), and thus, $C' \neq C$ is not a maximal clique of $\mathcal{G}$. This contradicts the fact that $\mathcal{G}'$ is a clique extension of $\mathcal{G}$. ■

The first property in Proposition 3.9 indicates that the elimination neighborhoods of $\mathcal{G}$ and $\mathcal{G}'$ are identical if a k-partial elimination ordering α down to C is chosen. The second property indicates that these elimination neighborhoods are cliques in their respective elimination graphs, or equivalently that α is also a k-partial elimination ordering for $\mathcal{G}$.

Hence, Proposition 3.8 shows that the graphs $\mathcal{G}$ and $\mathcal{G}'$ differ only on the subgraphs induced by C, and Proposition 3.9 shows that the graphs $\mathcal{G}$ and $\mathcal{G}'$ share the same perfect elimination orderings down to the clique C. We use these properties in what follows.

Probabilistic Relationships for Clique Extensions

The fact that the structure of a clique extension $\mathcal{G}'$ of a graph $\mathcal{G}$ differs only on the new maximal clique C suggests that perhaps the two projections $p_{\mathcal{G}}$ and $p_{\mathcal{G}'}$ differ only on their marginal densities $p_{\mathcal{G}}(x_C)$ and $p_{\mathcal{G}'}(x_C)$. We now show that this is the case by utilizing Proposition 3.9 and Corollary 3.1.

Let $\mathcal{G}$, $\mathcal{G}'$, C, and α be defined according to Proposition 3.9. Since α is a k-partial elimination ordering for both $\mathcal{G}$ and $\mathcal{G}'$, Corollary 3.1 suggests the following decompositions for $p_{\mathcal{G}'}$ and $p_{\mathcal{G}}$,

$$p_{\mathcal{G}'}(x) = p_{\mathcal{G}'}(x_C)\prod_{i=1}^{k} p\left(x_{\alpha(i)}|x_{N^{\downarrow}_{\mathcal{G}'}(\alpha(i))}\right) \tag{3.37a}$$

$$p_{\mathcal{G}}(x) = p_{\mathcal{G}}(x_C)\prod_{i=1}^{k} p\left(x_{\alpha(i)}|x_{N^{\downarrow}_{\mathcal{G}}(\alpha(i))}\right) = p_{\mathcal{G}}(x_C)\prod_{i=1}^{k} p\left(x_{\alpha(i)}|x_{N^{\downarrow}_{\mathcal{G}'}(\alpha(i))}\right), \tag{3.37b}$$

where the second equality in (3.37b) follows from the first part of Proposition 3.9. Since the product of conditional densities in (3.37a) and (3.37b) is identical, we can combine the two equations as follows,

$$p_{\mathcal{G}'}(x) = p_{\mathcal{G}'}(x_C)\prod_{i=1}^{k} p\left(x_{\alpha(i)}|x_{N^{\downarrow}_{\mathcal{G}'}(\alpha(i))}\right) = p_{\mathcal{G}'}(x_C)\left[\frac{p_{\mathcal{G}}(x)}{p_{\mathcal{G}}(x_C)}\right] = p_{\mathcal{G}}(x)\frac{p_{\mathcal{G}'}(x_C)}{p_{\mathcal{G}}(x_C)}.$$

Since C is a clique of $\mathcal{G}'$, we know that $p_{\mathcal{G}'}(x_C) = p(x_C)$, thereby giving the following important relationship

$$p_{\mathcal{G}'}(x) = p_{\mathcal{G}}(x)\frac{p(x_C)}{p_{\mathcal{G}}(x_C)}. \tag{3.38}$$

Notice that (3.38) can rewritten as $p_{\mathcal{G}'}(x) = p_{\mathcal{G}}(x_{V-C}|x_C)p(x_C)$, which indicates that the density $p_{\mathcal{G}}(x)$ can be transformed into the density $p_{\mathcal{G}'}(x)$ by keeping the same conditional density $p_{\mathcal{G}}(x_{V-C}|x_C)$ and by replacing the marginal $p_{\mathcal{G}}(x_C)$ with $p(x_C)$. Hence, the two densities truly differ only with respect to their marginals on the maximal clique C.

By defining the elimination graph $\mathcal{G}^{\downarrow} \triangleq \mathcal{G}\left\langle V - \{\alpha(1), \ldots, \alpha(k)\}\right\rangle$, (3.38) may also be written as follows,

$$p_{\mathcal{G}'}(x) = p_{\mathcal{G}}(x)\frac{p(x_C)}{p_{\mathcal{G}^{\downarrow}}(x)}.$$

This follows from Corollary 3.1 which indicates that $p_{\mathcal{G}^{\downarrow}}(x) = p_{\mathcal{G}}(x_C)$. Notice also that since α is a k-partial elimination ordering, Lemma 3.5 indicates that $\mathcal{G}^{\downarrow}$ is identical to the subgraph of $\mathcal{G}$ induced by $V - \{\alpha(1), \ldots, \alpha(k)\}$, which in this case is $\mathcal{G}\left(C\right)$. These facts are summarized in the following theorem.

Theorem 3.6 (Probabilistic Relationship for Clique Extensions).

Let $\mathcal{G}'$ be a clique extension of a triangulated graph $\mathcal{G} = (V, E)$, and suppose C is the additional maximal clique contained in $\mathcal{G}'$ but not $\mathcal{G}$. Define the induced subgraph $\mathcal{G}^{\downarrow} \triangleq \mathcal{G}(C)$. Then, for any density $p(x_V)$, the following probabilistic relationships hold,

$$p_{\mathcal{G}'}(x) = p_{\mathcal{G}}(x)\frac{p(x_C)}{p_{\mathcal{G}}(x_C)} = p_{\mathcal{G}}(x)\frac{p(x_C)}{p_{\mathcal{G}^{\downarrow}}(x)}. \tag{3.39}$$

Proof. See the preceding discussion. ■

The previous result can be extended to a sequence of graphs, where each graph is a clique extension of the preceding graph in the sequence. The following corollary states this fact.

Corollary 3.2 (Probabilistic Relationship for a Sequence of Clique Extensions).
Let $\mathcal{G} = (V, E)$ be a triangulated graph, and let $\mathcal{G} = \mathcal{G}_0, \mathcal{G}_1, \mathcal{G}_2, \ldots, \mathcal{G}_n = \mathcal{G}'$ be a sequence of graphs where $\mathcal{G}_i$ is a clique extension of $\mathcal{G}_{i-1}$ and where C_i is the unique maximal clique contained in $\mathcal{G}_i$ but not $\mathcal{G}_{i-1}$. Then, for any density $p(x_V)$, the following probabilistic relationships hold,

$$p_{\mathcal{G}'}(x) = p_{\mathcal{G}}(x)\prod_{i=1}^{n}\left[\frac{p(x_{C_i})}{p_{\mathcal{G}_{i-1}}(x_{C_i})}\right] = p_{\mathcal{G}}(x)\prod_{i=1}^{n}\left[\frac{p(x_{C_i})}{p_{\mathcal{G}_{i-1}(C_i)}(x)}\right]. \tag{3.40}$$

Proof. Follows directly by induction on (3.39). ■

Conditional Independencies and Clique Extensions

Using the functional relationship in (3.39), we now characterize the conditional independencies which a process X must satisfy in order for the two densities $p_{\mathcal{G}}$ and $p_{\mathcal{G}'}$ to be equal. In particular, $p_{\mathcal{G}} = p_{\mathcal{G}'}$ if and only if $p(x_C) = p_{\mathcal{G}^{\downarrow}}(x)$, and since $\mathcal{G}^{\downarrow}$ is a triangulated graph in Theorem 3.6, we know that $p(x_C) = p_{\mathcal{G}^{\downarrow}}(x)$ if and only if the subprocess X_C is Markov with respect to the graph $\mathcal{G}^{\downarrow}$. Hence, we can use (3.39) to determine precisely which conditional independencies guarantee that the two projections $p_{\mathcal{G}}$ and $p_{\mathcal{G}'}$ are equivalent, and these independencies only involve variables in the subprocess X_C.

For some choices of $\mathcal{G}$ and $\mathcal{G}'$, listing all of the conditional independencies implied by the subgraph $\mathcal{G}^{\downarrow}$ may be challenging. For example, for any triangulated graph $\mathcal{G}$, the complete graph is a clique extension with new maximal clique $C = V$. Using (3.39), the two densities p and $p_{\mathcal{G}}$ are equal if and only if X is Markov with respect to $\mathcal{G}$, but of course, we knew this already. Thus, the relationship in (3.39) provides little additional benefit in this particular case or similarly in cases where $\mathcal{G}$ and $\mathcal{G}'$ differ by a significant number of edges. One option for dealing with this issue is to consider a sequence of clique extensions such as in Corollary 3.2. By considering such a sequence, the problem of characterizing all conditional independencies implied by $\mathcal{G}$ may be split into smaller sub-problems, as we now show.

Given any triangulated graph $\mathcal{G}$ and any triangulated supergraph $\mathcal{G}'$, there always exists a sequence of triangulated graphs $\mathcal{G}_0 = \mathcal{G}, \mathcal{G}_1, \ldots, \mathcal{G}_n = \mathcal{G}'$ where each $\mathcal{G}_i$ has only one additional edge not contained in $\mathcal{G}_{i-1}$ [16]; this type of sequence is commonly called a *chordal sequence.* As the following lemma indicates, a chordal sequence always forms a sequence of clique extensions.

Lemma 3.6 (Chordal Sequences and Clique Extensions).
Let $\mathcal{G}_0 = \mathcal{G}, \mathcal{G}_1, \ldots, \mathcal{G}_n = \mathcal{G}'$ be a chordal sequence. Then, $\mathcal{G}_0 = \mathcal{G}, \mathcal{G}_1, \ldots, \mathcal{G}_n = \mathcal{G}'$ is also a sequence of clique extensions.

Proof. We only need to show that the case $n = 1$ holds, since the result follows directly from this case. Suppose $\mathcal{G}_0 = (V, E), \mathcal{G}_1 = (V, E')$ is a chordal sequence where $\{a, b\}$ is the added edge, *i.e.* $\{a, b\} \notin E$ and $\{a, b\} \in E'$. Suppose $\mathcal{G}_1$ is not a clique extension of $\mathcal{G}_0$, and hence, there exist at least two maximal cliques $C \neq C'$ contained in $\mathcal{G}_1$ but not $\mathcal{G}_0$. Since $\{a, b\}$ is the only added edge, $\{a, b\} \subseteq C$ and $\{a, b\} \subset C'$. If $\{a, b\} = C$ then there cannot exist a C' with $\{a, b\} \subset C'$ because C

would not be a maximal clique. Hence, $\{a, b\} \subset C$ and $\{a, b\} \subset C'$, and since $C \neq C'$, there must exists some $v \in C$ and $v' \in C'$ such that $\{v, v'\} \notin E'$ and thus $\{v, v'\} \notin E$. Then, $[v, a, v', b, v]$ is a cycle in $\mathcal{G}_0$ with no chord, which contradicts the fact that $\mathcal{G}_0$ is triangulated. ■

Therefore, for any two triangulated graphs $\mathcal{G}$ and $\mathcal{G}'$, where $\mathcal{G}'$ is a supergraph of $\mathcal{G}$, there always exists a sequence of clique extensions between $\mathcal{G}$ and $\mathcal{G}'$.

Given a sequence of clique extensions, (3.40) indicates that the constraints $p(x_{C_i}) = p_{\mathcal{G}_{i-1}(C_i)}(x)$, $i = 1, \ldots, n$, imply that $p_{\mathcal{G}'}$ and $p_{\mathcal{G}}$ are equal; however, it is not immediately obvious that the converse of this statement is true. In particular, if $p_{\mathcal{G}'} = p_{\mathcal{G}}$, the product of ratios on the right-hand side of (3.40) is equal to 1, but in general, this should not necessarily imply that each individual ratio is equal to 1. In this case, though, the fact that both $p_{\mathcal{G}}$ and $p_{\mathcal{G}'}$ are densities places some constraints on this relationship, and therefore, the ratios in (3.40) have special properties, as the following theorem indicates.

Theorem 3.7 (Conditional Independencies and Clique Extensions).
Let $\mathcal{G} = \mathcal{G}_0, \mathcal{G}_1, \mathcal{G}_2, \ldots, \mathcal{G}_n = \mathcal{G}'$ be a sequence of clique extensions with corresponding maximal cliques C_i as in Corollary 3.2. Then, for any density $p(x_V)$, the following are equivalent:

(1) $p_{\mathcal{G}'} = p_{\mathcal{G}}$,

(2) $p(x_{C_i}) = p_{\mathcal{G}_{i-1}}(x_{C_i}) = p_{\mathcal{G}_{i-1}(C_i)}(x)$ for $i = 1, \ldots, n$,

(3) X_{C_i} (under density p) is Markov with respect to the subgraph $\mathcal{G}_{i-1}(C_i)$ for $i = 1, \ldots, n$.

Proof. See Appendix B.4. ■

Based on the preceding discussion, the proof of Theorem 3.7 is, for the most part, straightforward. However, the equivalence between (1) and (2) is established by using the Kullback-Leibler divergence discussed in Section 3.10.1, and as such, we have placed the proof in Appendix B.4 for the interested reader.

Given two graphs $\mathcal{G}$ and $\mathcal{G}'$ as in Corollary 3.2, it is important to note that there may be several different sequences of clique extensions of the form $\mathcal{G} = \mathcal{G}_0, \mathcal{G}_1, \mathcal{G}_2, \ldots, \mathcal{G}_n = \mathcal{G}'$. Using Theorem 3.7, this implies that there are then several different ways to express the conditional independencies which equate $p_{\mathcal{G}}$ and $p_{\mathcal{G}'}$. The following example illustrates this idea as well as the other ideas discussed in this section.

Example 3.5 (Non-uniqueness of Sequences of Clique Extensions).
Consider the sequence of clique extensions $\mathcal{G}_0 = \mathcal{G}$, $\mathcal{G}_1$, $\mathcal{G}_2 = \mathcal{G}'$ shown in Figures 3.13(a), (b), and (c) respectively. For this sequence, $\mathcal{G}_1$ has a new maximal clique $C_1 = \{2, 4, 5, 6\}$, and $\mathcal{G}_2$ has a new maximal clique $C_2 = \{1, 2, 4, 5, 6\}$. Using Theorem 3.7, the two densities $p_{\mathcal{G}}$ and $p_{\mathcal{G}'}$ are equal if and only if the two constraints $p(x_{C_1}) = p_{\mathcal{G}_0(C_1)}(x)$ and $p(x_{C_2}) = p_{\mathcal{G}_1(C_2)}(x)$ are satisfied, where the subgraphs $\mathcal{G}_0(C_1)$ and $\mathcal{G}_1(C_2)$ are shown in Figures 3.13(d) and (e) respectively. Examining the structure of $\mathcal{G}_0(C_1)$ and $\mathcal{G}_1(C_2)$, notice that these two constraints imply the following factorizations and corresponding conditional independencies,

$$p(x_2, x_4, x_5, x_6) = p(x_4|x_2, x_5)p(x_6|x_2, x_5)p(x_2, x_5) \iff (X_4 \perp X_6) \,|\, (X_2, X_5) \qquad (3.41a)$$

$$p(x_1, x_2, x_4, x_5, x_6) = p(x_1|x_2, x_4)p(x_5, x_6|x_2, x_4)p(x_2, x_4) \iff (X_1 \perp (X_5, X_6)) \,|\, (X_2, X_4). \quad (3.41b)$$

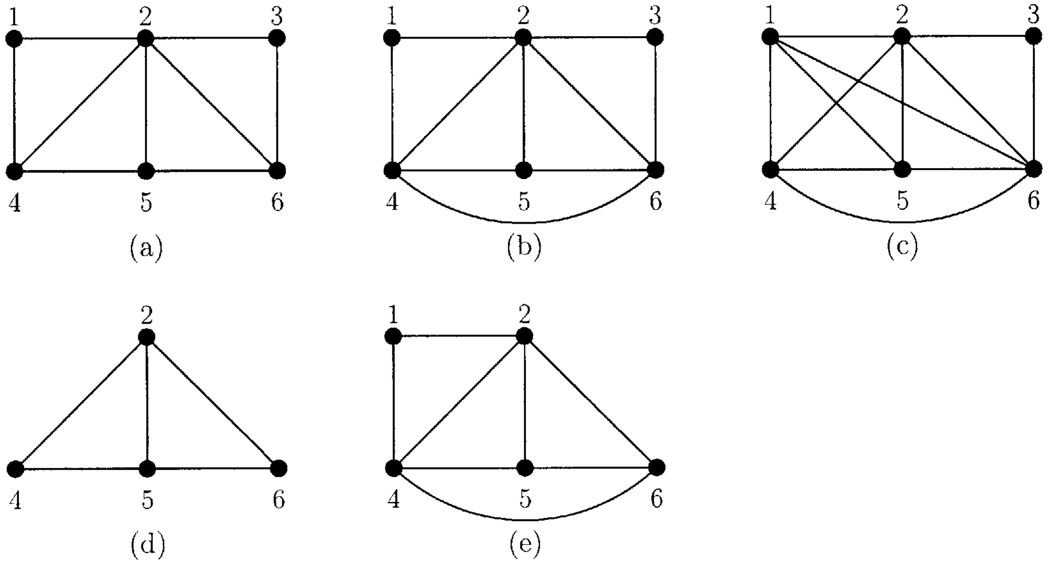


Figure 3.13. The sequence of clique extensions and corresponding subgraphs considered in Example 3.5. (a) A triangulated graph $\mathcal{G}_0$. (b) A clique extension $\mathcal{G}_1$ of $\mathcal{G}_0$ in (a) with new maximal clique $C_1 = \{2, 4, 5, 6\}$. (c) A clique extension $\mathcal{G}_2$ of $\mathcal{G}_1$ in (b) with new maximal clique $C_2 = \{1, 2, 4, 5, 6\}$. (d) The subgraph $\mathcal{G}_0(C_1)$. (e) The subgraph $\mathcal{G}_1(C_2)$.

For convenience, we use the notation $(X \perp Y) \,|Z$ to indicate X and Y are conditionally independent given Z.

Consider now a different sequence of clique extensions shown in Figures 3.14(a), (b), and (c), where the graphs $\mathcal{G}$ and $\mathcal{G}'$ are the same as before, but the intermediate graph $\mathcal{G}_1$ is different and has a new maximal clique $C_1 = \{1, 2, 4, 5\}$. The corresponding subgraphs $\mathcal{G}_0\,(C_1)$ and $\mathcal{G}_1\,(C_2)$, shown in Figures 3.14(d) and (e) respectively, are noticeably different from the subgraphs in Figures 3.13(d) and (e), and consequently, the implied conditional independencies differ from those given in (3.41),

$$p(x_1, x_2, x_4, x_5) = p(x_1|x_2, x_4)p(x_5|x_2, x_4)p(x_2, x_4) \iff (X_1 \perp X_5) \,|\, (X_2, X_4) \tag{3.42a}$$

$$p(x_1, x_2, x_4, x_5, x_6) = p(x_1, x_4|x_2, x_5)p(x_6|x_2, x_5)p(x_2, x_5) \iff ((X_1, X_4) \perp X_6) \,|\, (X_2, X_5)\,. \tag{3.42b}$$

However, as Theorem 3.7 indicates, the two sets of conditions in (3.41) and (3.42) are equivalent.

As this example illustrates, the choice of the sequence $\mathcal{G}_0 = \mathcal{G}, \mathcal{G}_1, \ldots, \mathcal{G}_n = \mathcal{G}'$ can significantly affect the list of conditional independencies stated in Theorem 3.7. This begs the question of what types of sequences lead to a "minimal" (given an appropriate definition of minimality) list of conditional independencies for given graphs $\mathcal{G}$ and $\mathcal{G}'$. We do not address this issue in this thesis, but instead, we later provide a recipe for generating an appropriate and useful sequence of clique extensions. ◀

■ 3.7.2 Neighborhood Separators

While the previous section derived a relationship between two densities $p_{\mathcal{G}}$ and $p_{\mathcal{G}'}$, this section focuses on the conditional independencies implied by special subgraphs of a graph $\mathcal{G}$. The graph-theoretic tool used to analyze these conditional independencies is called a *neighborhood separator*.

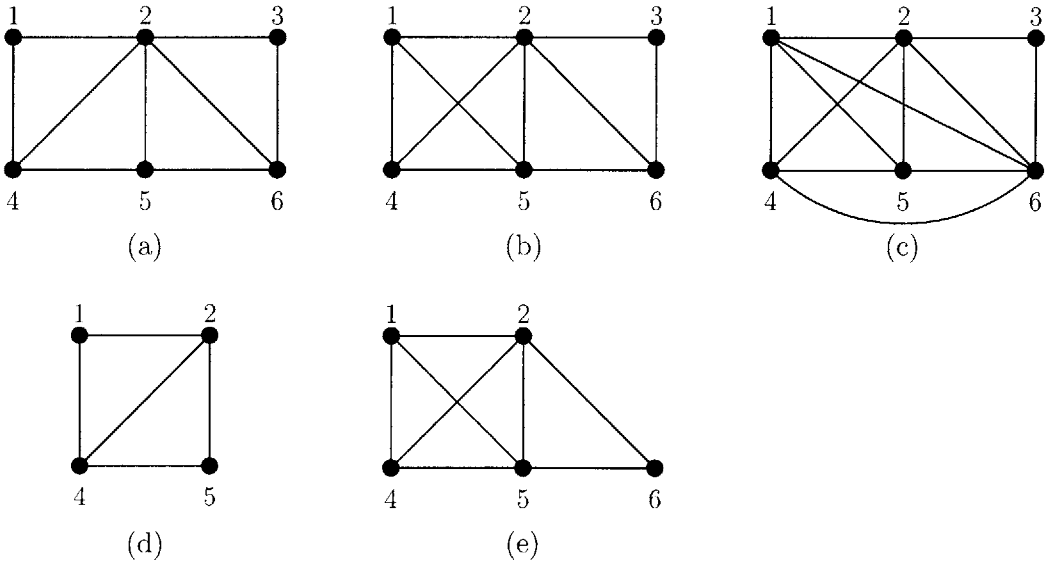


Figure 3.14. Another sequence of clique extensions and corresponding subgraphs considered in Example 3.5. (a) A triangulated graph $\mathcal{G}_0$. (b) A clique extension $\mathcal{G}_1$ of $\mathcal{G}_0$ in (a) with new maximal clique $C_1 = \{1, 2, 4, 5\}$. (c) A clique extension $\mathcal{G}_2$ of $\mathcal{G}_1$ in (b) with new maximal clique $C_2 = \{1, 2, 4, 5, 6\}$. (d) The subgraph $\mathcal{G}_0(C_1)$. (e) The subgraph $\mathcal{G}_1(C_2)$.

Definition 3.4 (Neighborhood Separators).
Given a graph $\mathcal{G} = (V, E)$, a *neighborhood separator* $S \subset V$ of $\mathcal{G}$ satisfies the following two properties:

(1) $N_{\mathcal{G}}[S] = N_{\mathcal{G}}[v]$ for all $v \in S$.

(2) The subgraph of $\mathcal{G}$ induced by $N_{\mathcal{G}}(S)$ contains $l \geq 1$ connected components induced by the vertices $C_1, \ldots, C_l$, where each C_i is a clique of $\mathcal{G}$. ◀

The first property of a neighborhood separator requires each vertex $v \in S$ to have exactly the same neighborhood as S, and this in turn implies that S must be a clique of $\mathcal{G}$. The second property indicates that if we define the subgraph $\mathcal{G}^{\downarrow} \triangleq \mathcal{G}\left(N_{\mathcal{G}}[S]\right)$, then S must be a separator of $\mathcal{G}^{\downarrow}$, *i.e.* S is a separator of its own neighborhood. Notice that this is a weaker condition than the usual notion of a separator in which S would have to separate the entire graph rather than simply its neighborhood. Notice also that the second property requires each of the connected components separated by S to be cliques; the importance of this requirement will become clear later.

For a graphical illustration of a neighborhood separator, consider the graph shown in Figure 3.15. In this example, vertex 5 is a neighborhood separator since it separates its neighborhood into the components $\{1, 2, 3\}$ and $\{7, 8\}$, each of which is a clique. However, vertex 5 is not a graph separator, since removing vertex 5 and all incident edges leaves a connected graph. The set $\{7, 8\}$ is also a neighborhood separator, since it separates its neighborhood into the trivial cliques $\{4\}$, $\{5\}$, and $\{6\}$.

The utility of neighborhood separators is derived from the fact that probability densities defined on subgraphs induced by neighborhood separators (and the conditional independencies associated with such a density) can be easily characterized. To see this, let $\mathcal{G}$ be a graph with neighborhood separator S and corresponding sets $C_1, \ldots, C_l$ as in Definition 3.4, and define the subgraph $\mathcal{G}^{\downarrow} \triangleq \mathcal{G}\left(N_{\mathcal{G}}[S]\right)$. Since S is a neighborhood separator, the maximal cliques of $\mathcal{G}^{\downarrow}$ are precisely the sets $C_i \cup S$, $i = 1, \ldots, l$, and consequently, we can immediately write the junction tree representation $\mathcal{T}^{\downarrow}$ of $\mathcal{G}^{\downarrow}$ as follows,

$$\mathcal{T}^{\downarrow} \triangleq (\mathcal{C}, \mathcal{S}), \text{ where } \mathcal{C} = \{C_i \cup S\}_{i=1}^{l}, \;\; \mathcal{S} = \{\underbrace{S, \ldots, S}_{(l-1) \text{ times}}\}. \tag{3.43}$$

One example of a junction tree for $\mathcal{G}^{\downarrow}$ is shown in Figure 3.16.

Using the junction tree representation in (3.43), any density p which factors according to the graph $\mathcal{G}^{\downarrow}$ may be written as follows,

$$p(x) = \frac{\prod_{i=1}^{l} p(x_{C_i}, x_S)}{p(x_S)^{l-1}} = p(x_S) \prod_{i=1}^{l} p(x_{C_i} | x_S). \tag{3.44}$$

The second equality in (3.44) shows the conditional independence structure associated with such a density p. Specifically, p factors according to (3.44) if and only if random vectors $X_{C_1}, \ldots, X_{C_l}$ are jointly independent conditioned on X_S, or equivalently, using the notation in Definition 2.5, these conditional independencies may be compactly stated as $\perp X_{\mathcal{C}}$ where $\mathcal{C}$ is the set of maximal cliques of $\mathcal{G}^{\downarrow}$. If some set C_i in Definition 3.4 is not a clique, then a process X which is Markov with respect to $\mathcal{G}^{\downarrow}$ would exhibit additional independencies than $\perp X_{\mathcal{C}}$, and for this reason, we require each C_i to be a clique.

Neighborhood separators are therefore useful in that they provide a means of accounting for the conditional independencies exhibited by a density defined on a special subgraph of a larger graph. By itself, such a characterization is not particularly useful, but when combined with the notion of a clique extension, these two graph-theoretic tools have significant implications for probabilistic modeling. To understand the significance, consider again the relationship in (3.39) between the two densities $p_{\mathcal{G}}$ and $p_{\mathcal{G}'}$. The nature of a clique extension allows these two densities to be functionally related by the ratio $p(x_C)/p_{\mathcal{G}^{\downarrow}}(x)$, where $\mathcal{G}^{\downarrow}$ is the subgraph of $\mathcal{G}$ induced by the maximal clique C. Since $p_{\mathcal{G}^{\downarrow}}$ is the projection of p onto the graph $\mathcal{G}^{\downarrow}$, this ratio can be written directly in terms of the marginals of p; however, the form of the density $p_{\mathcal{G}^{\downarrow}}$ depends on the structure of the subgraph $\mathcal{G}^{\downarrow}$, which for an arbitrary triangulated graph $\mathcal{G}$ may be a rather complicated function of p.

If C is equal to $N_{\mathcal{G}}[S]$ for some neighborhood separator S of $\mathcal{G}$, though, the density $p_{\mathcal{G}^{\downarrow}}(x)$ may be written in the form of (3.44), and consequently, the two densities $p_{\mathcal{G}'}$ and $p_{\mathcal{G}}$ are related in a simple and predictable way. Furthermore, Theorem 3.7 allows us to state the conditional independencies that guarantee $p_{\mathcal{G}'} = p_{\mathcal{G}}$. In particular, if $\mathcal{C}$ is the set of maximal cliques of $\mathcal{G}^{\downarrow}$ and if the conditions $\perp X_{\mathcal{C}}$ are satisfied with respect to the density p, then the ratio $p(x_C)/p_{\mathcal{G}^{\downarrow}}(x)$ is equal to 1 and $p_{\mathcal{G}'} = p_{\mathcal{G}}$.

Taking this idea one step further, consider the relationship in (3.40) for a sequence of clique extensions. In order to write the functional relationship between $p_{\mathcal{G}'}$ and $p_{\mathcal{G}}$ directly in terms of marginals of p, it is necessary to know the structure of each subgraph $\mathcal{G}_{i-1}(C_i)$. If by chance each C_i is equal to $N_{\mathcal{G}_{i-1}}[S_i]$ for some neighborhood separator S_i of $\mathcal{G}_{i-1}$, then the situation is the same as above, and the functional relationship is simple to characterize. For the sequence of graphs introduced in the next section, it is true that each C_i is equal to $N_{\mathcal{G}_{i-1}}[S_i]$ for some neighborhood separator S_i, but in order to prove that this property is satisfied, we need some

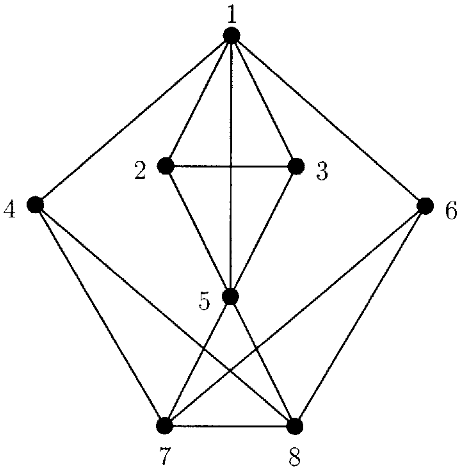


Figure 3.15. An example of a graph $\mathcal{G}$ which has a neighborhood separator covering. Specifically, the sets $\{1\}$,$\{2,3\}$,$\{4\}$,$\{5\}$,$\{6\}$, and $\{7,8\}$ form such a covering since each is a neighborhood separator of $\mathcal{G}$.

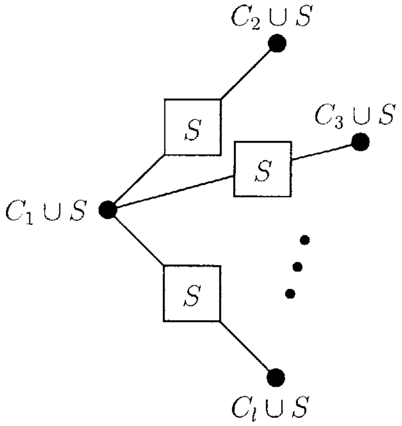


Figure 3.16. Given a graph $\mathcal{G}$ and a neighborhood separator S of $\mathcal{G}$, the figure shows one possible junction tree for the subgraph $\mathcal{G}^{\downarrow} \triangleq \mathcal{G}\left(N_{\mathcal{G}}[S]\right)$.

means of accounting for the neighborhood separators in a graph. For this purpose, we introduce the notion of a *neighborhood separator covering*.

Definition 3.5 (Neighborhood Separator Coverings).
Given a graph $\mathcal{G} = (V, E)$, a *neighborhood separator covering* is a collection $\{S_i\}_{i=1}^m$ of neighborhood separators S_i of $\mathcal{G}$ such that $V = \cup_{i=1}^m S_i$ and $S_i \cap S_j = \emptyset$ for all $i \neq j$. ◀

Therefore, a neighborhood separator covering is a collection of neighborhood separators which partition the vertices of a graph. Since this collection is not unique, though, a neighborhood separator covering does not provide an exhaustive list of all possible neighborhood separators for a given graph. For example, the graph shown in Figure 3.15 has neighborhood separators $\{1\}$,$\{2,3\}$,$\{4\}$,$\{5\}$,$\{6\}$, and $\{7,8\}$ which form a partition. However, vertices $\{2\}$ and $\{3\}$ are also neighborhood separators, and thus, the collection $\{\{1\},\{2\},\{3\},\{4\},\{5\},\{6\},\{7,8\}\}$ is also a neighborhood separator covering.

In addition to non-uniqueness, there are other interesting graph-theoretic questions concerning neighborhood separator coverings such as what types of graphs admit such a partitioning. For our purposes, it is not necessary to address these graph-theoretic questions since the graphs of interest to us have obvious neighborhood separator coverings. For example, consider the tree $\mathcal{G}_{\preceq}^{\sim}$ in Figure 3.4(b) as well as the augmented graph $\mathcal{G}_{\preceq}^{\sharp}$ in Figure 3.4(c). For the graph $\mathcal{G}_{\preceq}^{\sim}$, the individual vertices $\{v\}$ are the only neighborhood separators, and therefore, the collection $\{\{v\}\}_{v \in V}$ is the unique neighborhood separator covering. In Figure 3.4(e), the collection

$$\{\{0^{(t)}, 0^{(d)}\}, \{1^{(d)}\}, \{2^{(t)}, 2^{(d)}\}, \{3^{(t)}\}, \{4^{(t)}\}, \{5^{(t)}\}, \{6^{(t)}\}\}$$

is the unique neighborhood separator covering for $\mathcal{G}_{\preceq}^{\sharp}$.

Given a graph $\mathcal{G}$ with a neighborhood separator covering $\{S_i\}_{i=1}^m$, such as $\mathcal{G}_{\preceq}^{\sim}$ and $\mathcal{G}_{\preceq}^{\sharp}$ for example, we now examine how two different graphical operations affect this covering. As the following proposition indicates, when $\mathcal{G}$ is triangulated, the process of adding the edges in $D_{\mathcal{G}}(v)$ (for any $v \in V$) to $\mathcal{G}$ does not destroy the neighborhood separator covering, *i.e.* $\{S_i\}_{i=1}^m$ is a neighborhood separator covering for the resulting graph.

Proposition 3.10 (Neighborhood Separator Coverings and Adding Edges).
Let $\mathcal{G} = (V, E)$ be a triangulated graph with neighborhood separator covering $\{S_i\}_{i=1}^m$. Given some $v \in V$, define $\mathcal{G}' \triangleq (V, E \cup D_{\mathcal{G}}(v))$. Then, $\{S_i\}_{i=1}^m$ is also a neighborhood separator covering for $\mathcal{G}'$.

Proof. See Appendix B.5. ■

Using the preceding proposition, the following corollary indicates that a neighborhood separator covering changes in a predictable way when a vertex v is eliminated from a triangulated graph. Specifically, the covering remains unchanged except that vertex v is removed from the set S_i containing v.

Corollary 3.3 (Neighborhood Separator Coverings and Vertex Elimination).
Let $\mathcal{G} = (V, E)$ be a triangulated graph with neighborhood separator covering $\{S_i\}_{i=1}^m$. Given some $v \in V$ and $v \in S_i$, define $\mathcal{G}^{\downharpoonleft} \triangleq \downharpoonleft(\mathcal{G}, v)$. Then, $\{S_1, \ldots, S_{i-1}, S_i - \{v\}, S_{i+1}, \ldots, S_m\}$ is a neighborhood separator covering for $\mathcal{G}^{\downharpoonleft}$.

Proof. See Appendix B.5. ■

Since a neighborhood separator covering changes in a predictable fashion for a single vertex elimination, it changes in similarly predictable fashion for a sequence of vertex eliminations. Consequently, given an initial triangulated graph $\mathcal{G}$ with a neighborhood separator covering $\{S_i\}_{i=1}^m$, we can keep track of the neighborhood separator covering as vertices are sequentially eliminated from the graph $\mathcal{G}$. This fact, along with the simple factorization in (3.44), allows us to explicitly state the functional relationship in (3.40) directly in terms of marginals of p. We demonstrate this process in the next section for a specific sequence of graphs.

■ 3.8 The Modified Elimination Game

In Section 3.6, vertex elimination and the elimination game were introduced. This section discusses a useful modification of the elimination game and shows how the graphs generated by the modified elimination game relate to clique extensions and neighborhood separators. Section 3.8.1 introduces the modified elimination game, and Section 3.8.2 provides an important graph-theoretic characterization of the graphs generated by the modified elimination game. In Section 3.8.3, the graph-theoretic results and probabilistic relationships discussed in the previous section for general triangulated graphs are applied to the graphs generated by the modified elimination game.

■ 3.8.1 Definition and Notation

The modified elimination game is similar to the elimination game in the sense that a sequence of graphs are generated by successively adding edges and removing vertices. The modification is that at every step in the sequence, the current vertex may or may not be removed from the graph. More specifically, let $\mathcal{G} = (V, E)$ with $|V| = n$ be an arbitrary graph with an associated ordering α on V, and let $M \subseteq V$ be a specified subset of the vertices.[22] The *modified elimination graphs* are defined as follows,[23]

$$\widetilde{\mathcal{G}}_0^{\downarrow} \triangleq (V_0, F_0) = \mathcal{G} = (V, E) \tag{3.45a}$$

$$\widetilde{\mathcal{G}}_i^{\downarrow} \triangleq (V_i, F_i) = \begin{cases} \left(V_{i-1}, F_{i-1} \cup D_{\widetilde{\mathcal{G}}_{i-1}^{\downarrow}}(\alpha(i))\right) & \text{if } \alpha(i) \in M, \\ \Big\downarrow\left(\widetilde{\mathcal{G}}_{i-1}^{\downarrow}, \alpha(i)\right) & \text{if } \alpha(i) \notin M, \end{cases} \qquad i = 1, \dots, n. \tag{3.45b}$$

To better understand (3.45), recall that eliminating a vertex v from a graph $\mathcal{G}$ has two steps: (1) edges are added to $\mathcal{G}$ in order to make $N_{\mathcal{G}}(v)$ a clique and (2) vertex v and all incident edges are removed from the graph. In (3.45b), if $\alpha(i) \notin M$ the graph $\widetilde{\mathcal{G}}_i^{\downarrow}$ is obtained from $\widetilde{\mathcal{G}}_{i-1}^{\downarrow}$ by eliminating vertex $\alpha(i)$ from the graph, *i.e.* both steps of vertex elimination are performed. On the other hand, if $\alpha(i) \in M$ then only the first step of vertex elimination is performed, *i.e.* edges are added to $\widetilde{\mathcal{G}}_{i-1}^{\downarrow}$ so that $N_{\widetilde{\mathcal{G}}_{i-1}^{\downarrow}}(\alpha(i))$ becomes a clique in $\widetilde{\mathcal{G}}_i^{\downarrow}$. A graphical illustration of this procedure is shown in Figures 3.17 and 3.18 for two different orderings of the vertices but the same set $M = \{2\}$. Comparing Figures 3.9 and 3.17 (and similarly Figures 3.10 and 3.18), notice that the modified elimination game generates more fill edges than the elimination game; this is of course due to the fact that vertex 2 is never removed from the graph.

[22] For now, the set M is an arbitrary set with no specific meaning, but in Section 3.9, this same M will be interpreted as the marginalization constraint set discussed before.

[23] The dependence on the set M and the ordering α is suppressed for notational clarity.

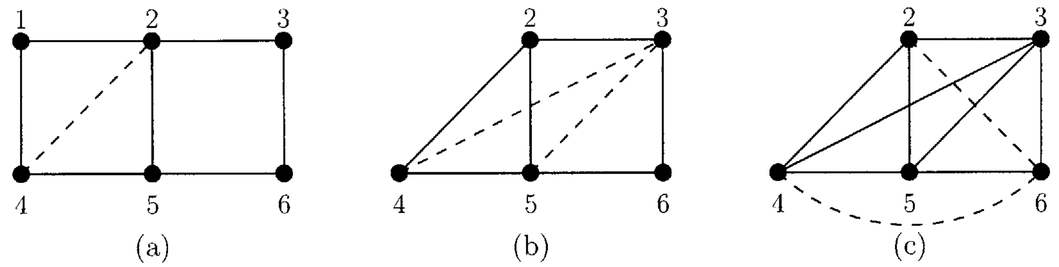


Figure 3.17. Graphical illustration of the first three graphs in a sequence of modified elimination graphs for the vertex ordering $\alpha = (1,2,3,4,5,6)$ and $M = \{2\}$. The dashed edges indicate the modified elimination deficiencies. (a) $\tilde{\mathcal{G}}_0^{\downarrow} = \mathcal{G}$ (solid), $\tilde{D}_{\mathcal{G}}^{\downarrow}(1)$ (dashed) (b) $\tilde{\mathcal{G}}_1^{\downarrow}$ (solid), $\tilde{D}_{\mathcal{G}}^{\downarrow}(2)$ (dashed) (c) $\tilde{\mathcal{G}}_2^{\downarrow}$ (solid), $\tilde{D}_{\mathcal{G}}^{\downarrow}(3)$ (dashed)

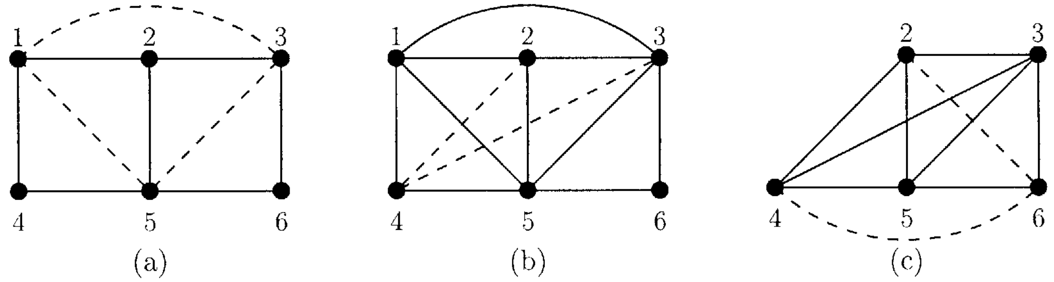


Figure 3.18. Graphical illustration of the first three graphs in a sequence of modified elimination graphs for the vertex ordering $\alpha = (2,1,3,4,5,6)$ and $M = \{2\}$. The dashed edges indicate the modified elimination deficiencies. (a) $\tilde{\mathcal{G}}_0^{\downarrow} = \mathcal{G}$ (solid), $\tilde{D}_{\mathcal{G}}^{\downarrow}(2)$ (dashed) (b) $\tilde{\mathcal{G}}_1^{\downarrow}$ (solid), $\tilde{D}_{\mathcal{G}}^{\downarrow}(1)$ (dashed) (c) $\tilde{\mathcal{G}}_2^{\downarrow}$ (solid), $\tilde{D}_{\mathcal{G}}^{\downarrow}(3)$ (dashed)

The fact that the modified elimination game generates more fill edges than the elimination game is the reason why we consider this modification; the importance of these additional edges will become clear later. For now, we introduce notation, similar to that of the elimination game, to represent the edges added during the modified elimination game. In particular, given an ordering α, the *modified elimination deficiency* of vertex $v = \alpha(i)$ is defined as

$$\tilde{D}_{\mathcal{G}}^{\downarrow}(v) \triangleq D_{\tilde{\mathcal{G}}_{i-1}^{\downarrow}}(v), \quad v = \alpha(i), \tag{3.46}$$

i.e. it is the deficiency of vertex $\alpha(i)$ in the modified elimination graph $\tilde{\mathcal{G}}_{i-1}^{\downarrow}$. Similarly, the *modified elimination neighborhood* of vertex $v = \alpha(i)$ is defined as

$$\tilde{N}_{\mathcal{G}}^{\downarrow}(v) \triangleq N_{\tilde{\mathcal{G}}_{i-1}^{\downarrow}}(v), \quad v = \alpha(i), \tag{3.47a}$$

$$\tilde{N}_{\mathcal{G}}^{\downarrow}[v] \triangleq \tilde{N}_{\mathcal{G}}^{\downarrow}(v) \cup \{v\}. \tag{3.47b}$$

As an example, the dashed lines in Figures 3.17(a),(b), and (c) indicate the edges contained in $\tilde{D}_{\mathcal{G}}^{\downarrow}(1)$, $\tilde{D}_{\mathcal{G}}^{\downarrow}(2)$, and $\tilde{D}_{\mathcal{G}}^{\downarrow}(3)$ respectively, for the ordering $\alpha = (1,2,3,4,5,6)$ and $M = \{2\}$. The dashed lines in Figures 3.18(a),(b), and (c) indicate the edges contained in $\tilde{D}_{\mathcal{G}}^{\downarrow}(2)$, $\tilde{D}_{\mathcal{G}}^{\downarrow}(1)$, and $\tilde{D}_{\mathcal{G}}^{\downarrow}(3)$ respectively, for the ordering $\alpha = (2,1,3,4,5,6)$ and $M = \{2\}$.

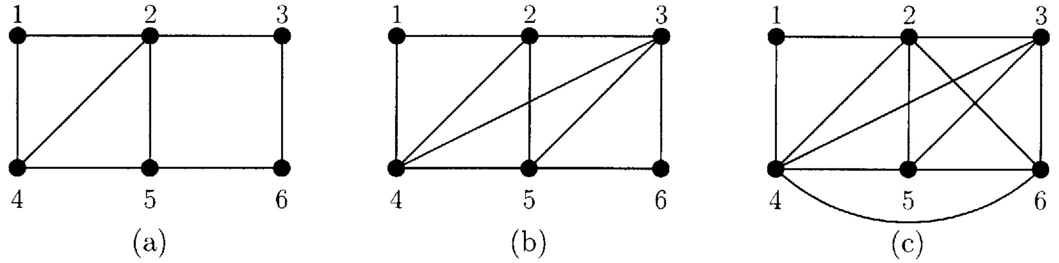


Figure 3.19. Graphical illustration of the first three graphs in the sequence $\widetilde{\mathcal{G}}_i$ in (3.48) assuming the initial graph $\widetilde{\mathcal{G}}_0 = \mathcal{G}$ shown in Figure 3.17(a) (solid), the vertex ordering $\alpha = (1, 2, 3, 4, 5, 6)$, and $M = \{2\}$. (a) $\widetilde{\mathcal{G}}_1$ (b) $\widetilde{\mathcal{G}}_2$ (c) $\widetilde{\mathcal{G}}_3$

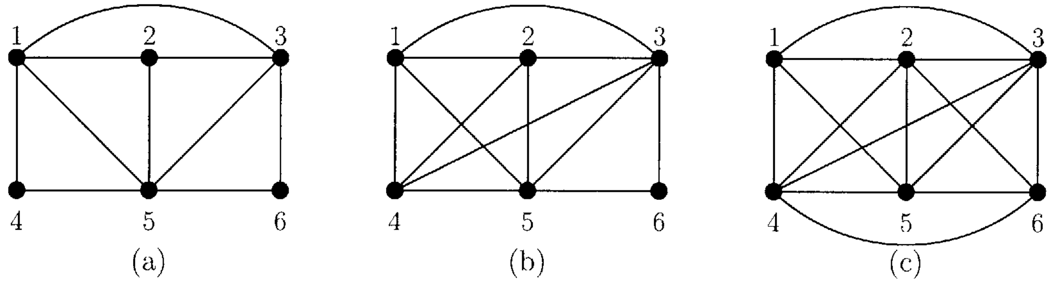


Figure 3.20. Graphical illustration of the first three graphs in the sequence $\widetilde{\mathcal{G}}_i$ in (3.48) assuming the initial graph $\widetilde{\mathcal{G}}_0 = \mathcal{G}$ shown in Figure 3.18(a) (solid), the vertex ordering $\alpha = (2, 1, 3, 4, 5, 6)$, and $M = \{2\}$. (a) $\widetilde{\mathcal{G}}_1$ (b) $\widetilde{\mathcal{G}}_2$ (c) $\widetilde{\mathcal{G}}_3$

Using the fill edges generated at each step of the modified elimination game, we now introduce a new sequence of graphs which form the basis of our discussion in the remainder of this section. Specifically, given a graph $\mathcal{G} = (V, E)$ with $|V| = n$, an ordering α, and a set $M \subseteq V$, we define the following graphs

$$\widetilde{\mathcal{G}}_0 \triangleq (V, E_0) = \mathcal{G} = (V, E) \tag{3.48a}$$

$$\widetilde{\mathcal{G}}_i \triangleq (V, E_i) = \left(V, E_{i-1} \cup \widetilde{D}^{\downarrow}_{\mathcal{G}}(\alpha(i))\right), \quad i = 1, \ldots, n. \tag{3.48b}$$

Notice that this sequence differs from the modified elimination graphs in the sense that each graph is defined on the same set of vertices. In particular, (3.48) defines a sequence where $\widetilde{\mathcal{G}}_i$ is a supergraph of $\widetilde{\mathcal{G}}_{i-1}$, differing only on the edges in the set $\widetilde{D}^{\downarrow}_{\mathcal{G}}(\alpha(i))$.

Figure 3.19 provides an example of the graphs $\widetilde{\mathcal{G}}_i$, assuming the initial graph $\widetilde{\mathcal{G}}_0 = \mathcal{G}$ is represented by the solid lines in Figure 3.17(a) with $\alpha = (1, 2, 3, 4, 5, 6)$ and $M = \{2\}$. This sequence is generated by successively adding the dashed lines shown in Figure 3.17. Similarly, Figure 3.20 provides the same type of example for the ordering $\alpha = (2, 1, 3, 4, 5, 6)$ and $M = \{2\}$.

Notice that while the elimination graphs $\widetilde{\mathcal{G}}^{\downarrow}_2$ shown in Figures 3.17(c) and 3.18(c) are the same, the graphs $\widetilde{\mathcal{G}}_3$ shown in Figures 3.19(c) and 3.20(c) are different. This is due to the fact that the fill edges $\widetilde{D}^{\downarrow}_{\mathcal{G}}(v)$, $v \in V$, vary greatly depending on the ordering α. To better understand the graphs $\widetilde{\mathcal{G}}_i$, we now provide an important characterization of the edges contained in each such graph.

■ 3.8.2 Characterization of Edges

In the previous section, we showed how to construct the graphs $\widetilde{\mathcal{G}}_i$ in (3.48) by recursively adding the edges generated by the modified elimination game. In this section, we characterize the edges contained in the graph $\widetilde{\mathcal{G}}_i$ solely in terms of the initial graph $\mathcal{G}$, the ordering α, and the set M, without reference to the sequence of steps involved in the modified elimination game. This characterization is a generalization of the characterization given in [90] for the elimination game, and it is useful for proving some of the remaining graph-theoretic results.

To simplify the characterization provided below in Proposition 3.11, we introduce a function β^{-1} defined in terms of a specified ordering α. Given a set V with $|V| = n$, a set $M \subseteq V$, and an ordering α on V, we define $\beta^{-1} : V \to \{1, \dots, n, \infty\}$ as follows,

$$\beta^{-1}(v) \triangleq \begin{cases} \alpha^{-1}(v) & \text{if } v \notin M, \\ \infty & \text{if } v \in M. \end{cases} \tag{3.49}$$

Intuitively, β^{-1} maintains the same ordering as α for the vertices not in the set M and assigns ∞ to all of the vertices in M.

Proposition 3.11 (Edges Associated with the Modified Elimination Game).
Let $\mathcal{G} = (V, E)$ be an arbitrary graph, and let $\widetilde{\mathcal{G}}_i = (V, E_i)$ be defined according to (3.48) for some ordering α and some $M \subseteq V$. Define β^{-1} according to (3.49). Then, $\{a, b\} \in E_i$ if and only if there exists a path $[a = v_1, v_2, \dots, v_k, v_{k+1} = b]$ in $\mathcal{G}$ such that $\alpha^{-1}(v_j) < \min(\beta^{-1}(a), \beta^{-1}(b), i+1)$, for $j = 2, \dots, k$.[24]

Proof. See Appendix B.6. ■

For an illustration of this characterization, consider the graph $\widetilde{\mathcal{G}}_3$ shown in Figure 3.19(c). Notice that edge $\{4, 6\}$ is present because the path $[4, 1, 2, 3, 6]$ in $\mathcal{G}$ ($\mathcal{G}$ is shown in Figure 3.17(a)) satisfies the requirements of Proposition 3.11. On the other hand, edge $\{1, 5\}$ is not present in $\widetilde{\mathcal{G}}_3$ shown in Figure 3.19(c) because any path between 1 and 5 in $\mathcal{G}$ must contain either vertex 4 or vertex 2, and neither of these vertices satisfy the requirements of Proposition 3.11.

One immediate consequence of Proposition 3.11 is that it suggests an important property of the graph $\widetilde{\mathcal{G}}_n$, *i.e.* the final graph in the sequence. Namely, if $\mathcal{G}$ is connected then M is a clique of $\widetilde{\mathcal{G}}_n$. To see this, choose any $\{a, b\} \subseteq M$, and notice that by definition $\beta^{-1}(a) = \infty$ and $\beta^{-1}(b) = \infty$. Since $\mathcal{G}$ is connected, there must be a path from a to b in $\mathcal{G}$, and any such path trivially satisfies the conditions of Proposition 3.11. Consequently, $\{a, b\} \in E_n$, thereby proving that M is a clique of $\widetilde{\mathcal{G}}_n$. This property of the modified elimination game is the reason we chose to introduce such a modification and the reason why more fill edges are desirable. We return to this idea in Section 3.9.

■ 3.8.3 Tying It All Together

In this section, we provide three propositions which relate the graph-theoretic results and probabilistic relationships presented in Section 3.7 to the graphs $\widetilde{\mathcal{G}}_i$ generated by the modified elimination game. The first proposition indicates that the graphs $\widetilde{\mathcal{G}}_i$ in (3.48) form a sequence of clique extensions if the initial graph $\widetilde{\mathcal{G}}_0 = \mathcal{G}$ is triangulated, and furthermore, the corresponding maximal cliques C_i are equal to the modified elimination neighborhoods $\widetilde{N}^{\downarrow}_{\mathcal{G}}[\alpha(i)]$ in (3.47b). The result follows from induction and application of Proposition 3.7.

[24] If $\{a, b\} \in E$, then we assume that $k = 1$, and consequently, these conditions are trivially satisfied.

Proposition 3.12 (Modified Elimination Game and Clique Extensions).
Suppose $\mathcal{G} = (V, E)$ is a triangulated graph. Given any set $M \subseteq V$ and any ordering α on the vertices V, the sequence of graphs $\widetilde{\mathcal{G}}_i$ in (3.48) form a sequence of clique extensions, and the new maximal clique C_i contained in $\widetilde{\mathcal{G}}_i$ but not $\widetilde{\mathcal{G}}_{i-1}$ is given by $C_i = \widetilde{N}^{\downarrow}_{\mathcal{G}}[\alpha(i)]$.

Proof. See Appendix B.7. ■

Notice that while the graphs $\widetilde{\mathcal{G}}_i$ form a sequence of clique extensions, they do not necessarily correspond to a chordal sequence since more than one edge may be added to $\widetilde{\mathcal{G}}_{i-1}$ in order to form $\widetilde{\mathcal{G}}_i$. Using the graphs $\widetilde{\mathcal{G}}_i$, we can always construct a chordal sequence $\{\mathcal{H}_i\}$ such that $\{\widetilde{\mathcal{G}}_i\}$ is a subsequence of $\{\mathcal{H}_i\}$, but within the context of the multiscale realization problem, this additional step is not necessary.

As previously shown in Proposition 3.10 and Corollary 3.3, a neighborhood separator covering of a triangulated graph changes in a predictable way under two simple graph operations. As the following proposition indicates, these two results may be applied in a sequential fashion to prove that the elimination graphs $\widetilde{\mathcal{G}}^{\downarrow}_i$ have a neighborhood separator covering, assuming that the initial graph $\widetilde{\mathcal{G}}_0 = \mathcal{G}$ is triangulated and has a neighborhood separator covering.

Proposition 3.13 (Modified Elimination Game and Neighborhood Separators).
Suppose $\mathcal{G} = (V, E)$ is a triangulated graph with a neighborhood separator covering. Consider the sequence of graphs $\widetilde{\mathcal{G}}^{\downarrow}_i$ in (3.45) for a given set $M \subseteq V$ and ordering α on V. For $i = 0, \ldots, n$, each graph $\widetilde{\mathcal{G}}^{\downarrow}_i$ has a neighborhood separator covering.

Proof. We prove the result by induction. By assumption $\widetilde{\mathcal{G}}^{\downarrow}_0 = \mathcal{G}$ is triangulated and has a neighborhood separator covering. Using Proposition 3.12, $\widetilde{\mathcal{G}}^{\downarrow}_{i-1}$ is triangulated, and by the induction hypothesis, $\widetilde{\mathcal{G}}^{\downarrow}_{i-1}$ has a neighborhood separator covering. If $\alpha(i) \in M$, then $\widetilde{\mathcal{G}}^{\downarrow}_i$ is obtained from $\widetilde{\mathcal{G}}^{\downarrow}_{i-1}$ by adding the edges in $D_{\widetilde{\mathcal{G}}^{\downarrow}_{i-1}}(\alpha(i))$, and using Proposition 3.10, $\widetilde{\mathcal{G}}^{\downarrow}_i$ has the same neighborhood separator covering as $\widetilde{\mathcal{G}}^{\downarrow}_{i-1}$. If $\alpha(i) \notin M$, then $\widetilde{\mathcal{G}}^{\downarrow}_i = \Big\downarrow\left(\widetilde{\mathcal{G}}^{\downarrow}_{i-1}, \alpha(i)\right)$, and by Corollary 3.3, $\widetilde{\mathcal{G}}^{\downarrow}_i$ has a neighborhood separator covering. ■

Notice that the results stated in Proposition 3.10 and Corollary 3.3 are stronger than needed for the preceding proposition. In fact, Proposition 3.10 and Corollary 3.3 may be used to show how an initial neighborhood separator covering $\{S_i\}_{i=1}^{m}$ for the graph $\mathcal{G}$ changes in each of the modified elimination graphs $\widetilde{\mathcal{G}}^{\downarrow}_i$. Such a characterization is not needed for our purposes since we are only interested in the fact that a neighborhood separator covering exists.

The final result provided in this section is the following restatement of Theorem 3.7 for the graphs $\widetilde{\mathcal{G}}_i$.

Proposition 3.14 (Modified Elimination Game and Theorem 3.7).
Suppose $\mathcal{G} = (V, E)$ is a triangulated graph with a neighborhood separator covering. Let the sequence of graphs $\widetilde{\mathcal{G}}_i$ be defined according to (3.48) for some set $M \subseteq V$ and an ordering α on V. Let $\mathcal{C}_i$ denote the set of maximal cliques of the subgraph $\widetilde{\mathcal{G}}_{i-1}(C_i)$, where $C_i \triangleq \widetilde{N}^{\downarrow}_{\mathcal{G}}[\alpha(i)]$ for $i = 1 \ldots n$. Then, for any density $p(x_V)$, the following are equivalent:

(1) $p_{\widetilde{\mathcal{G}}_i} = p_{\mathcal{G}}$ for some $1 \leq i \leq n$,

(2) the conditions $\perp X_{\mathcal{C}_j}$ are satisfied (under density p) for all $1 \leq j \leq i$.

Proof. See Appendix B.7. ■

Therefore, when the initial graph $\widetilde{\mathcal{G}}_0 = \mathcal{G}$ is triangulated and has a neighborhood separator covering, the graphs $\widetilde{\mathcal{G}}_i$ generated by the modified elimination game have sufficient structure to permit an important simplification to the statement in Theorem 3.7. In particular, Proposition 3.13 indicates that the elimination graphs $\widetilde{\mathcal{G}}_i^{\downarrow}$ have a neighborhood separator covering, and consequently, each maximal clique C_i is equal to $N_{\widetilde{\mathcal{G}}_{i-1}^{\downarrow}}[S]$ for some neighborhood separator S of $\widetilde{\mathcal{G}}_{i-1}^{\downarrow}$. As a result, each subgraph $\widetilde{\mathcal{G}}_{i-1}(C_i)$ has the simple structure discussed in Section 3.7.2 for neighborhood separators, and a process X_{C_i}, which is Markov with respect to $\widetilde{\mathcal{G}}_{i-1}(C_i)$, exhibits only the conditional independencies $\perp X_{C_i}$. Using this simplification, we are now in a position to prove the unproven results provided in Chapter 2.

3.9 Marginalization-Invariant Markovianity Revisited

In this section, we clarify the relationship between the marginalization-invariant Markov property introduced in the previous chapter and the sufficient conditions for exact realization presented in this chapter, and in establishing this relationship, we immediately prove Theorems 2.3 and 2.4. Recall from Section 2.6 that the marginalization-invariant Markov property is a set-theoretic characterization of the Markov properties required for exact realization. In contrast, this chapter presents a graph-theoretic characterization of the sufficient conditions required for exact realization. The following section shows that these two characterizations are equivalent for the multiscale realization problem, and Section 3.9.2 shows their equivalence for the state augmentation problem.

3.9.1 Sufficient Conditions for Exact Multiscale Realization

The theoretical results presented in this chapter provide a powerful framework for characterizing and enumerating the set of conditional independencies sufficient to solve the exact multiscale realization problem. In particular, Theorem 3.3 suggests a sufficient condition in order for a density $p \in \mathcal{P}^M(V, d)$ to be a solution to problem $\mathcal{P}^M$, which then leads to the solution p^T of the exact realization problem $\mathcal{Q}$. This sufficient condition requires $p_\mathcal{G} = p_{\mathcal{G}_{\precsim}}$ for a particular class of graphs $\mathcal{G}$. Theorem 3.7 later provides a convenient graph-theoretic approach for listing the conditional independencies that a process X must satisfy in order for its density to satisfy $p_\mathcal{G} = p_{\mathcal{G}_{\precsim}}$. Of course, this list depends on the particular sequence of clique extensions which is chosen.

Section 3.8 suggests a method for creating a sequence of clique extensions given an initial triangulated graph and an ordering on the vertices, and furthermore, this sequence is special in that the final graph in the sequence is guaranteed to have a clique equal to M. Given a sequence of clique extensions generated by the modified elimination game, Proposition 3.14 states that the conditional independencies associated with such a sequence, as established by Theorem 3.7, may be characterized in a simple manner as long as the initial graph is triangulated and has a neighborhood separator covering. Since the tree $\mathcal{G}_{\precsim}$ is triangulated and has a neighborhood separator covering, we can then use this theoretical framework to state a sufficient set of conditional independencies for solutions to the exact realization problem, and such a characterization is given in Proposition 3.15 to follow.

As noted earlier, one interesting property of the graphs $\widetilde{\mathcal{G}}_i$ generated by the modified elimination game is that the final graph in the sequence has a clique equal to M – a property which must be satisfied in order for the sufficient conditions in Theorem 3.3 to apply. However, if the initial graph $\widetilde{\mathcal{G}}_0$ is equal to the tree $\mathcal{G}_{\widetilde{\preceq}}$, then some orderings on the vertices lead to an intermediate graph $\widetilde{\mathcal{G}}_i$ with a clique equal to M. This fact is evidenced by the following lemma.

Lemma 3.7 (Modified Elimination Graphs and Trees).
Let $\mathcal{G}_{\preceq} = (V, E)$ be a specified rooted tree, and let α be any ordering on V such that for all non-leaf vertices $v \in V$, $\alpha^{-1}(v) \leq m$ for some $1 \leq m \leq |V|$. Define the sequence of graphs $\widetilde{\mathcal{G}}_i$ according to (3.48) for some set $M \subseteq V$, the initial graph $\widetilde{\mathcal{G}}_0 = \mathcal{G}_{\widetilde{\preceq}}$, and the ordering α. Then, $\widetilde{\mathcal{G}}_m$ has a clique equal to M.

Proof. Choose any $\{a, b\} \subseteq M$. According to Proposition 3.11, $\{a, b\}$ is an edge in the graph $\widetilde{\mathcal{G}}_m$ if and only if there exists a path $[a = u_1, u_2, \ldots, u_k, u_{k+1} = b]$ in $\mathcal{G}_{\widetilde{\preceq}}$ such that $\alpha^{-1}(u_j) < \min(\beta^{-1}(a), \beta^{-1}(b), m + 1)$ for $2 \leq j \leq k$. Since $a, b \in M$, we have $\beta^{-1}(a) = \beta^{-1}(b) = \infty$, and thus, the vertices u_j need only satisfy $\alpha^{-1}(u_j) < m + 1$. Using the fact that $\mathcal{G}_{\widetilde{\preceq}}$ is a tree, each u_j, $2 \leq j \leq k$, must be a non-leaf vertex, and by the definition of α, the condition $\alpha^{-1}(v) < m + 1$ is satisfied for all non-leaf vertices of $\mathcal{G}_{\widetilde{\preceq}}$. Hence, $\{a, b\}$ is an edge in $\widetilde{\mathcal{G}}_m$. ■

Therefore, the intermediate graph $\widetilde{\mathcal{G}}_m$ has a clique equal to M as long as the ordering α has only leaf vertices after vertex $\alpha(m)$. Because of this fact, we choose to only consider orderings α which place all non-leaf vertices first and all leaf vertices last, and we call such an ordering a *leaf-last ordering.*

Using this fact and the results presented so far in this chapter, we can now provide a list of the conditional independencies which ensure a solution to the exact realization problem. In the following proposition, we assume that the graphs $\widetilde{\mathcal{G}}_i$ are defined according to (3.48) with the initial graph $\widetilde{\mathcal{G}}_0 = \mathcal{G}_{\widetilde{\preceq}}$, and as in Proposition 3.14, we let $\mathcal{C}_i$ denote the maximal cliques of $\widetilde{\mathcal{G}}_{i-1}$ (C_i), where $C_i \triangleq \widetilde{N}^{\downarrow}_{\mathcal{G}_{\widetilde{\preceq}}}[\alpha(i)]$.

Proposition 3.15 (Sufficient Conditional Independencies for Exact Realization).
Let $\mathcal{G}_{\preceq}$ be a rooted tree defined on vertex set V with m non-leaf vertices, and let α be any leaf-last ordering on V. Let $p^*(x_M)$ be a given target density for some $M \subseteq V$. If a random process $\{X_v\}_{v \in V}$ with density $p(x_V)$ satisfies the conditional independencies $\perp X_{\mathcal{C}_i}$ for $1 \leq i \leq m$, then $p^T(x_M) = p(x_M)$, and if $p \in \mathcal{P}^M(V, d)$, then p^T is solution to problem $\mathcal{Q}$.

Proof. Follows directly from Lemma 3.7, Proposition 3.14, and Theorem 3.3. ■

Notice that the result given in Proposition 3.15 resembles the result stated previously in Theorem 2.3. The only difference is that Theorem 2.3 considers the conditions $\perp X_{\mathcal{M}_{v_i}}$ associated with the marginalization-invariant Markov property, whereas Proposition 3.15 considers the conditions implied by the sets of maximal cliques $\mathcal{C}_i$. As we show in Proposition 3.16, these conditions are equivalent and related in a simple way.

To develop some intuition, notice that the sets $\mathcal{M}_{v_i}$ and $\mathcal{C}_i$ differ in a very distinctive way. In particular, the maximal cliques contained in $\mathcal{C}_i$ have larger cardinality with increasing values of i since the graphs $\widetilde{\mathcal{G}}_i$ contain more and more edges. In contrast, the sets contained in $\mathcal{M}_{v_i}$ have smaller cardinality with increasing values of i because of the fact that successive boundary sets B_{v_i} have smaller cardinality. In fact, we can show that the collection of sets $\mathcal{M}_{v_i}$ and $\mathcal{C}_i$, $i = 1, \ldots, m$, are the same but simply permuted.

To formalize this relationship, let $(v_1, \ldots, v_m)$ be any ordering on the non-leaf vertices of $\mathcal{G}_{\preceq} = (V, E)$,[25] and let α be any leaf-last ordering on V such that

$$\alpha(m - i + 1) = v_i, \quad i = 1, \ldots, m, \tag{3.50}$$

i.e. $\alpha(1) = v_m, \alpha(2) = v_{m-1}, \ldots, \alpha(m) = v_1$. Therefore, α flips the ordering $(v_1, \ldots, v_m)$ and places all leaf vertices last. Given such an ordering, the following proposition shows that the sets $\mathcal{M}_{v_i}$ and $\mathcal{C}_{m-i+1}$ are equal for $i = 1, \ldots, m$.

Proposition 3.16 (Marginalization-Invariant Markovianity and Proposition 3.15).
Let $(v_1, \ldots, v_m)$ be an ordering on the non-leaf vertices of a rooted tree $\mathcal{G}_{\preceq} = (V, E)$, and let α be any leaf-last ordering on V satisfying (3.50). Assume that $M \subseteq V$ contains all leaf vertices of $\mathcal{G}_{\preceq}$, and let $\mathcal{C}_i$ be defined as in Proposition 3.15. Then, $\mathcal{C}_{m-i+1} = \mathcal{M}_{v_i}$ for $i = 1, \ldots, m$.

Proof. See Appendix B.8. ■

Notice that the preceding proposition requires M to contain all leaf vertices of $\mathcal{G}_{\preceq}$. This condition is necessary in order to ensure the equality of $\mathcal{C}_{m-i+1}$ and $\mathcal{M}_{v_i}$, but it is not a restriction since the leaf vertices of the tree can always be pruned until this condition is satisfied.

The result in Proposition 3.16 thereby proves that the two sets of conditions $\{\perp X_{\mathcal{M}_{v_i}}\}_{1 \leq i \leq m}$ and $\{\perp X_{\mathcal{C}_i}\}_{1 \leq i \leq m}$ are equivalent, and combining this fact with Proposition 3.15 proves Theorem 2.3. Consequently, we now have two equivalent sets of sufficient conditions for the exact realization problem. The conditions $\perp X_{\mathcal{C}_i}$ are derived from a graph-theoretic scheme to account for the conditional independencies in a sequence of graphs. In contrast, the conditions $\perp X_{\mathcal{M}_{v_i}}$ are based on a set-theoretic scheme of intersecting the conditional independencies required by the global Markov property with the marginalization constraint set M. Both types of conditions provide valuable insights into the exact realization problem, and both lead to a sequential realization procedure for designing exact multiscale models, as discussed previously in Section 2.7.2.

■ 3.9.2 Sufficient Conditions for Exact Multiscale Realization Using Augmented States

With very little additional work, we can generalize the results stated in the previous section to the class of multiscale models with augmented states. In particular, we generalize Lemma 3.7, Proposition 3.15, and Proposition 3.16 in what follows.

Recall from Section 3.3.3 that a special augmented graph $\mathcal{G}^{\sharp}_{\preceq}$ may be constructed from a rooted tree $\mathcal{G}_{\preceq} = (V, E)$ and a set $M \subseteq V$, and this augmented graph provides a convenient means of indexing the augmented states of a multiscale model. As discussed in Section 3.4.4, we choose to redefine the set M according to (3.24) so that it only contains the target vertices $v^{(t)} \in V^{\sharp}$. For this redefined set M, we generalize Lemma 3.7 to show that certain orderings α on $V^{\sharp}$ lead to an intermediate graph $\widetilde{\mathcal{G}}_m$ with a clique equal to M.

Lemma 3.8 (Modified Elimination Graphs and $\mathcal{G}^{\sharp}_{\preceq}$).
Let $\mathcal{G}_{\preceq} = (V, E)$ be a specified rooted tree, and let $\mathcal{G}^{\sharp}_{\preceq} = (V^{\sharp}, E^{\sharp})$ be the corresponding augmented graph for some set $M \subseteq V$. Redefine M according to (3.24), and let α be any ordering on $V^{\sharp}$ such that for any $v^{(d)} \in V^{\sharp}$, $\alpha^{-1}(v^{(d)}) \leq m$ for some $1 \leq m \leq |V^{\sharp}|$. Define the sequence of graphs $\widetilde{\mathcal{G}}_i$

[25] Recall that we considered such an ordering $(v_1, \ldots, v_m)$ in the previous chapter when we were only interested in ordering the non-leaf vertices.

according to (3.48) for the redefined set M, the initial graph $\widetilde{\mathcal{G}}_0 = \mathcal{G}^{\sharp}_{\preceq}$, and the ordering α. Then, $\widetilde{\mathcal{G}}_m$ has a clique equal to M.

Proof. The proof is similar to that in Lemma 3.7. Namely, we must show that there exists a path $[a = u_1, u_2, \ldots, u_k, u_{k+1} = b]$ in $\mathcal{G}^{\sharp}_{\preceq}$ such that $\alpha^{-1}(u_j) < m+1$ for $2 \leq j \leq k$. Because of the structure of $\mathcal{G}^{\sharp}_{\preceq}$, we can construct a path between any vertices $a, b \in V^{\sharp}$ which passes only through design vertices $v^{(d)}$, and by the definition of α, the condition $\alpha^{-1}(u_j) < m+1$ is then satisfied. ∎

Because of this fact, we choose to only consider orderings α which place all design vertices of $\mathcal{G}^{\sharp}_{\preceq}$ first and all target vertices last, and we call such an ordering a *target-last ordering.*

For the following generalization of Proposition 3.15, assume that a rooted tree $\mathcal{G}_{\preceq} = (V, E)$ is given along with a target density $p^*(x_M)$ defined on some set $M \subseteq V$, and let $\mathcal{G}^{\sharp}_{\preceq} = (V^{\sharp}, E^{\sharp})$ be the corresponding augmented graph. As in Section 3.4.4, redefine the set M according to (3.24), and let $p^*(x_M)$ be indexed by this new set M. Finally, let the graphs $\widetilde{\mathcal{G}}_i$ be defined according to (3.48) for the initial graph $\widetilde{\mathcal{G}}_0 = \mathcal{G}^{\sharp}_{\preceq}$, the modified set M, and an ordering α on $V^{\sharp}$, and as in Proposition 3.14, let $\mathcal{C}_i$ denote the maximal cliques of $\widetilde{\mathcal{G}}_{i-1}\,(C_i)$, where $C_i \triangleq \widetilde{N}^{\downarrow}_{\mathcal{G}^{\sharp}_{\preceq}}[\alpha(i)]$.

Proposition 3.17 (Sufficient Conditional Independencies for Exact Realization Using Augmented States).

Let α be any target-last ordering on $V^{\sharp}$, where $V^{\sharp}$ contains m design vertices $v^{(d)}$. If a random process $\{X_v\}_{v \in V}$ with density $p(x_V)$ satisfies the conditional independencies $\perp X_{\mathcal{C}_i}$ for $1 \leq i \leq m$, then $p^T(x_M) = p(x_M)$, and if $p \in \mathcal{P}^M(V, d)$, then p^T is solution to problem $\mathcal{Q}$.

Proof. Follows directly from Lemma 3.8, Proposition 3.14, and Theorem 3.4. ∎

Finally, we generalize Proposition 3.16 to the case of augmented states by showing that the maximal cliques $\mathcal{C}_i$ in Proposition 3.17 and the augmented marginalization-invariant families $\mathcal{M}^{\sharp}_{v_i}$ are simply permuted versions of one another. To formalize this relationship, let $(v_1, \ldots, v_m)$ be any ordering on the non-leaf vertices of $\mathcal{G}_{\preceq} = (V, E)$, and let α be any target-last ordering on $V^{\sharp}$ such that

$$\alpha(m - i + 1) = v_i^{(d)}, \quad i = 1, \ldots, m. \tag{3.51}$$

Therefore, α is of the form $\alpha = (v_m^{(d)}, v_{m-1}^{(d)}, \ldots, v_1^{(d)}, \ldots)$ with all of the target vertices following the design vertices. Given such an ordering, the following proposition shows that the sets $\mathcal{M}^{\sharp}_{v_i}$ and $\mathcal{C}_{m-i+1}$ are equal for $i = 1, \ldots, m$.

Proposition 3.18 (Marginalization-Invariant Markovianity and Proposition 3.17).

Let $(v_1, \ldots, v_m)$ be an ordering on the non-leaf vertices of a rooted tree $\mathcal{G}_{\preceq} = (V, E)$, and let α be any target-last ordering on $V^{\sharp}$ satisfying (3.51). If C_i is defined as in Proposition 3.17, then, $\mathcal{C}_{m-i+1} = \mathcal{M}^{\sharp}_{v_i}$ for $i = 1, \ldots, m$.

Proof. See Appendix B.9. ∎

Notice that the preceding proposition does not explicitly constrain the initial set M to contain the leaf vertices of $\mathcal{G}_{\preceq}$, as was required in Proposition 3.16. This is because we imposed this constraint already in Section 3.3.3 when defining the augmented graph $\mathcal{G}^{\sharp}_{\preceq}$.

The result in Proposition 3.18 thereby proves that the two sets of conditions $\{\perp X_{\mathcal{M}^{\sharp}_{v_i}}\}_{1 \leq i \leq m}$ and $\{\perp X_{\mathcal{C}_i}\}_{1 \leq i \leq m}$ are equivalent. Combining this fact with Proposition 3.17 proves Theorem 2.4.

3.10 Approximate Multiscale Realization: A Relaxed Problem Formulation

Having discussed the exact multiscale realization problem in detail, we now return to the approximate multiscale realization problem introduced in Section 3.1. As we show, many of the ideas presented for the exact realization problem directly translate into similar ideas for this relaxed version of the realization problem. Given some measure of discrimination $D(p\|q)$ between two densities p and q, recall from Section 3.1 that the approximate multiscale realization problem is defined as follows:

> Approximate Multiscale Realization Problem $\tilde{\mathcal{Q}}$: Find any density $\hat{q} \in \mathcal{P}_{\mathcal{G}_{\preceq}}(V, d)$ which minimizes the cost $D(p^*(x_M)\|\hat{q}(x_M))$, *i.e.*
>
> $$\hat{q} = \arg \min_{q \in \mathcal{P}_{\mathcal{G}_{\preceq}}(V,d)} D(p^*(x_M)\|q(x_M)).$$

This relaxed formulation of the exact realization problem $\mathcal{Q}$ is henceforth referred to as *approximate multiscale realization problem* $\tilde{\mathcal{Q}}$.

In this thesis, we focus on a particularly important and useful choice for the functional $D(\cdot\|\cdot)$ – namely, the *Kullback-Leibler divergence* discussed in Section 3.10.1. In Section 3.10.2, we introduce an important mapping which is an integral part of the approximate realization problem. In Section 3.10.3, we suggest a series of alternative problems to problem $\tilde{\mathcal{Q}}$, and we discuss necessary conditions for solving each such alternative. Finally, in Section 3.10.4 we provide an important decomposition of the cost $D(p^*(x_M)\|q(x_M))$ associated with problem $\tilde{\mathcal{Q}}$.

3.10.1 Kullback-Leibler Divergence As a Measure of Approximation

The *Kullback-Leibler (KL) divergence* [69] has found great utility in the area of information theory [17, 42, 68] due to its close relationship to the notions of entropy and mutual information and utility in the area of information geometry [1, 2, 4, 18, 20] due to its in interesting geometric properties. For our purposes, the KL divergence is not interpreted as a measure of information but rather a cost function, providing a measure of discrimination between two densities p and q, and as we discuss here, KL has several important properties which we use extensively in subsequent sections.

Given two probability densities $p(x)$ and $q(x)$ defined on a continuous space $\mathcal{X}$, the KL divergence is given by the following,

$$D(p(x)\|q(x)) \triangleq \int_{\mathcal{X}} p(x) \log\left(\frac{p(x)}{q(x)}\right) dx, \tag{3.52}$$

and if the space $\mathcal{X}$ is discrete, the integral is replaced by a summation. Using Jensen's inequality, it can be shown that $D(p\|q) \geq 0$ for all p and q, with equality if and only if $p = q$ almost everywhere. Therefore, the KL divergence provides a measure of discrimination between two densities p and q, with large values of $D(p\|q)$ indicating that p and q are dissimilar. However, $D(p\|q)$ is not symmetric in p and q and does not necessarily satisfy the triangle inequality, and as a result, it is not a true distance measure.

In using the KL divergence as a measure of discrimination, we encounter an important technical consideration not encountered in the exact realization problem. Specifically, the divergence in (3.52) may not be defined for some choice of densities p and q. For a discrete space $\mathcal{X}$, the divergence is undefined when $p(x) > 0$ and $q(x) = 0$ for some $x \in \mathcal{X}$, and for a continuous space $\mathcal{X}$, the divergence is undefined when $\int_{\mathcal{X}_R} p(x)dx > 0$ and $\int_{\mathcal{X}_R} q(x)dx = 0$ for some measurable set$\mathcal{X}_R \subset \mathcal{X}$. In order to simplify subsequent discussion, we henceforth assume (but do not explicitly state) that all densities p and q under consideration satisfy the technical conditions necessary for (3.52) to be defined.

Given two probability densities $p(x, y)$ and $q(x, y)$ defined on a continuous space $\mathcal{X} \times \mathcal{Y}$, the KL divergence may also be used to provide a measure of discrimination between two conditional densities $p(y|x)$ and $q(y|x)$. Using the definition in (3.52), though, the divergence $D\left(p(y|x) \| q(y|x)\right)$ depends on the particular instantiation of $X = x$ and is therefore a function over the space $\mathcal{X}$. For convenience, we define the functional $\bar{D}\left(p(y|x) \| q(y|x)\right)$ to be the average of $D\left(p(y|x) \| q(y|x)\right)$ over all instantiations of $X = x$ as follows,[26]

$$\begin{aligned} \bar{D}\left(p(y|x) \| q(y|x)\right) &\triangleq \int_{\mathcal{X}} p(x) D\left(p(y|x) \| q(y|x)\right) dx \\ &= \int_{\mathcal{X}} \int_{\mathcal{Y}} p(x, y) \log\left(\frac{p(y|x)}{q(y|x)}\right) dy dx. \end{aligned} \tag{3.53}$$

Using (3.52) and (3.53), it can be shown that KL satisfies the following additivity property [17],

$$D\left(p(x, y) \| q(x, y)\right) = D\left(p(x) \| q(x)\right) + \bar{D}\left(p(y|x) \| q(y|x)\right). \tag{3.54}$$

Another important property of KL is that the triangle inequality is satisfied (with equality) for some choices of densities. For example, the following decomposition indicates that the divergence between two densities p and q is equal to the sum of the divergences $D\left(p(x) \| p_{\mathcal{G}}(x)\right)$ and $D\left(p_{\mathcal{G}}(x) \| q(x)\right)$ if $q(x)$ factors according to a subgraph of $\mathcal{G}$.

Proposition 3.19 (Additivity of Projections).
Let $\mathcal{H} = (V, E)$ be any graph defined on a vertex set V, and let $\mathcal{G} = (V, E')$ be any triangulated supergraph of $\mathcal{H}$, i.e. *$E' \supseteq E$. Suppose $p(x_V)$ and $q(x_V)$ are two densities indexed by V and defined on the same space $\mathcal{X} = \prod_{v \in V} \mathcal{X}_v$. If $q(x)$ factors according to $\mathcal{H}$, then the following decomposition holds,*

$$D\left(p(x) \| q(x)\right) = D\left(p(x) \| p_{\mathcal{G}}(x)\right) + D\left(p_{\mathcal{G}}(x) \| q(x)\right). \tag{3.55}$$

Proof. See Appendix B.10. ■

Using the decomposition in (3.55), we can also state an important geometric property of $p_{\mathcal{G}}$. Namely, $p_{\mathcal{G}}$ is the "closest" density to p (in the KL sense) which factors according to $\mathcal{G}$, or said in a different way, $p_{\mathcal{G}}$ is the "best" model for p amongst all densities with the conditional independencies implied by $\mathcal{G}$. This fact establishes the appropriateness of calling $p_{\mathcal{G}}$ a projection of p onto $\mathcal{G}$ and is stated in the following corollary to Proposition 3.19.

Corollary 3.4 ($p_{\mathcal{G}}$ is a KL Projection).
Let $\mathcal{G} = (V, E)$ be a triangulated graph, and let $p(x_V)$ be indexed by V and defined on the space $\mathcal{X} = \prod_{v \in V} \mathcal{X}_v$. Then, $q(x) = p_{\mathcal{G}}(x)$ is a minimizer of $D\left(p(x) \| q(x)\right)$ over all $q(x_V)$ defined on $\mathcal{X}$ and satisfying $q = q_{\mathcal{G}}$.[27]

[26] Some authors define this average measure to be the KL divergence between conditional densities.

[27] We cannot say that $q(x) = p_{\mathcal{G}}(x)$ is the unique minimizer when $\mathcal{X}$ is a continuous space.

Proof. Let $\mathcal{H} = \mathcal{G}$ in Proposition 3.19. Then, the first term in (3.55) is constant with respect to $q(x)$, and the second term can be minimized by setting $q(x) = p_{\mathcal{G}}(x)$. ■

One particularly important illustration of the result in Corollary 3.4 occurs when $\mathcal{G} = \mathcal{G}_{\preceq}^{\sim}$ is a tree. In this case, $q(x) = p^T(x)$ is the minimizer of $D\left(p(x) \| q(x)\right)$ over all multiscale densities $q(x)$ with the structure of $\mathcal{G}_{\preceq}$. Comparing this optimization problem to problem $\widetilde{\mathcal{Q}}$, we see that the two problems are identical when $M = V$; that is, when the target density is $p^*(x_M) = p^*(x_V)$, a closed-form solution to problem $\widetilde{\mathcal{Q}}$ is given by $\hat{q} = (p^*)^T$. However, when $M \subset V$, the problem is not so trivial, and this is the topic we address in the remaining sections of this chapter.

■ 3.10.2 An Important Mapping

In this section, we introduce an important mapping with a number of special properties. Specifically, for a fixed target density $p^*(x_M)$, we define the mapping $\mathcal{F}^M$ as follows,

$$\mathcal{F}^M : \mathcal{P}(V,d) \longrightarrow \mathcal{P}^M(V,d), \quad \mathcal{F}^M(q) \triangleq q(x_{V-M}|x_M)p^*(x_M). \tag{3.56}$$

This mapping transforms any $q \in \mathcal{P}(V,d)$ to a density $\bar{p}(x) \triangleq \mathcal{F}^M(q(x))$ with the desired marginal $\bar{p}(x_M) = p^*(x_M)$, while maintaining the same conditional density $\bar{p}(x_{V-M}|x_M) = q(x_{V-M}|x_M)$.

Since $\mathcal{F}^M(\cdot)$ only alters the marginal density of X_M, the KL divergence between $\bar{p}(x)$ and $q(x)$ is equal to the divergence between $p^*(x_M)$ and $q(x_M)$, a fact that can be shown by using the additivity property (3.54),

$$\begin{aligned} D\left(\mathcal{F}^M(q(x)) \| q(x)\right) &= D\left(\bar{p}(x) \| q(x)\right) = D\left(\bar{p}(x_M) \| q(x_M)\right) + \bar{D}\left(\bar{p}(x_{V-M}|x_M) \| q(x_{V-M}|x_M)\right) \\ &= D\left(p^*(x_M) \| q(x_M)\right). \end{aligned} \tag{3.57}$$

Using this relationship, the divergence $D\left(p^*(x_M) \| q(x_M)\right)$ considered in problem $\widetilde{\mathcal{Q}}$ can equivalently be written as a divergence between the two densities $\mathcal{F}^M(q(x))$ and $q(x)$, each defined on the entire space $\mathcal{X}$ rather than the subspace $\mathcal{X}_M$.

Geometric Properties of the Mapping $\mathcal{F}^M(\cdot)$

As we now discuss, the pair of densities q and $\mathcal{F}^M(q)$ have an interesting geometric relationship. In particular, the following proposition indicates that the KL divergence satisfies the triangle inequality (with equality) for any three densities $p \in \mathcal{P}^M(V,d)$, $q \in \mathcal{P}(V,d)$, and $\mathcal{F}^M(q) \in \mathcal{P}^M(V,d)$.

Proposition 3.20 (The Geometry of the Mapping $\mathcal{F}^M(\cdot)$).
Let $p \in \mathcal{P}^M(V,d)$ and $q \in \mathcal{P}(V,d)$. Then, the following decomposition holds,

$$D\left(p(x) \| q(x)\right) = D\left(p(x) \| \mathcal{F}^M(q(x))\right) + D\left(\mathcal{F}^M(q(x)) \| q(x)\right). \tag{3.58}$$

Proof. Follows directly from the definition of KL, the relationship in (3.57), and the fact that $p \in \mathcal{P}^M(V,d)$. See Appendix B.10 for details. ■

Notice that the density $\mathcal{F}^M(q)$ in (3.58) plays a role similar to the projection $p_{\mathcal{G}}$ in (3.55). In fact, as the following corollary to Proposition 3.20 indicates, $\mathcal{F}^M(q)$ is the projection of q onto the set $\mathcal{P}^M(V,d)$. The two projections $p_{\mathcal{G}}$ and $\mathcal{F}^M(q)$ are fundamentally different, though, in the sense that $q = p_{\mathcal{G}}$ is a minimizing solution with respect to the second argument of $D(\cdot\|\cdot)$, while $p = \mathcal{F}^M(q)$ is a minimizing solution with respect to the first argument of $D(\cdot\|\cdot)$.

Corollary 3.5 ($\mathcal{F}^M(q)$ is a KL Projection).
Given any $q \in \mathcal{P}(V,d)$, the density $\hat{p}(x) = \mathcal{F}^M(q(x))$ is a minimizer of the function $D\left(p(x)\|q(x)\right)$ over all $p \in \mathcal{P}^M(V,d)$.

Proof. The second term in (3.58) is a function of $q(x)$ and does not depend on $p(x)$, and the first term can be minimized by setting $p(x) = \mathcal{F}^M(q(x))$. ■

For the class of exponential models [4, 12], the density $p_{\mathcal{G}}$ is called an M-projection, while the density $\mathcal{F}^M(q)$ is called an E-projection [1, 2]. The geometric properties of this particular family of models have been well-studied in the literature [12, 18, 28, 57, 104], and a number of optimization algorithms such as the famous EM algorithm may be viewed as performing an alternating series of projections [19, 20, 59].

Multiscale Models and the Mapping $\mathcal{F}^M(\cdot)$

Consider now the implications of Corollary 3.5 when q has additional conditional independence structure, and in particular, suppose q is a multiscale density, *i.e.* $q \in \mathcal{P}_{\mathcal{G}_{\preceq}}(V,d)$. Corollary 3.5 indicates that $\mathcal{F}^M(q)$ is the "closest" density to q which lies in the set $\mathcal{P}^M(V,d)$, or said differently, $\mathcal{F}^M(q)$ is the "closest" density with the target marginal $p^*(x_M)$. However, since $\mathcal{F}^M(q)$ has the marginal $p^*(x_M)$, it is not necessarily a multiscale density, and for most choices of $p^*(x_M)$, $\mathcal{F}^M(q)$ possesses fewer conditional independencies than q.

To see this, suppose $p^*(x_M)$ possesses no conditional independencies and is therefore Markov with respect to the complete graph on M, denoted by $K_M \triangleq (M, E_M)$, and further suppose that $\mathcal{C}$ is the set of all maximal cliques of $\mathcal{G}_{\preceq}^{\sim} = (V,E)$. We now show that $\mathcal{F}^M(q)$ factors according to the graph $\mathcal{G} \triangleq (V, E \cup E_M)$, *i.e.* the graph containing the edges in the tree $\mathcal{G}_{\preceq}^{\sim}$ plus the edges in the complete graph K_M. Using the fact that $q(x)$ factors according to $\mathcal{G}_{\preceq}^{\sim}$ for some choice of compatibility functions ψ_C, the density $\mathcal{F}^M(q)$ may be written as follows

$$\mathcal{F}^M(q(x)) = q(x_{V-M}|x_M)p^*(x_M) = q(x)\frac{p^*(x_M)}{q(x_M)} = \left[\prod_{C \in \mathcal{C}} \psi_C(x_C)\right] \psi_M(x_M), \tag{3.59}$$

where $\psi_M(x_M) \triangleq \frac{p^*(x_M)}{q(x_M)}$. Consequently, $\mathcal{F}^M(q(x))$ may be written as a product of compatibility functions over a set of cliques $\mathcal{C} \cup \{M\}$, and since this set is necessarily a subset of the maximal cliques of the graph $\mathcal{G}$, $\mathcal{F}^M(q(x))$ factors according to $\mathcal{G}$. As an example, Figures 3.21(a) and (b) respectively show a tree $\mathcal{G}_{\preceq}^{\sim} = (V,E)$ and the graph $\mathcal{G} = (V, E \cup E_M)$, where $M = \{3,4,5,6\}$.

In summary, the density $\mathcal{F}^M(q)$ may possess fewer conditional independencies than q (depending on $p^*(x_M)$) because it necessarily factors according to a graph which is fully connected on the vertices M. This fact is important for subsequent discussion because it proves that the set $\mathcal{P}_{\mathcal{G}}^M(V,d)$ is non-empty for every graph $\mathcal{G}$ (triangulated or not triangulated) which is a supergraph of $\mathcal{G}_{\preceq}^{\sim}$ and has a clique equal to M. This follows from the fact that every density $p \in \mathcal{P}(V,d)$ has a corresponding multiscale density p^T which maps to $\mathcal{F}^M(p^T)$, and since $\mathcal{F}^M(p^T)$ factors according to the graph $\mathcal{G} = (V, E \cup E_M)$, it also factors according to every supergraph of $\mathcal{G}$.

An Invariance Property

The mappings $\mathcal{F}^M(\cdot)$ and $p \longrightarrow p^T$ exhibit an interesting property when applied to solutions $\hat{q}$ of problem $\widetilde{\mathcal{Q}}$. In particular, the mapping $\left(\mathcal{F}^M(\cdot)\right)^T$, *i.e.* the composition of the two mappings $\mathcal{F}^M(\cdot)$

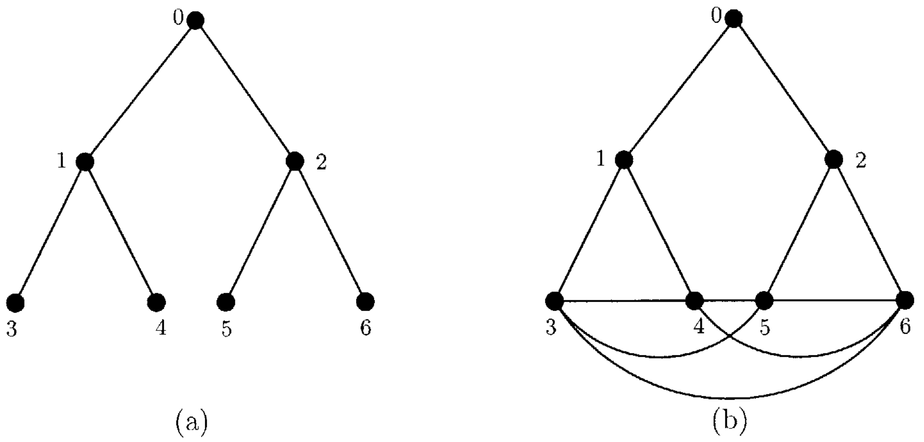


Figure 3.21. (a) A tree $\mathcal{G}_{\precsim} = (V, E)$. (b) A graph $\mathcal{G}$ containing the edges E in $\mathcal{G}_{\precsim}$ plus the edges needed to make $M = \{3, 4, 5, 6\}$ a clique. If q is a density that factors according to the graph in (a), then $\mathcal{F}^M(q)$ factors according to $\mathcal{G}$.

and $p \longrightarrow p^T$, is idempotent on the solution set of problem $\widetilde{\mathcal{Q}}$, and therefore, every solution $\hat{q}$ satisfies $\hat{q} = \left(\mathcal{F}^M(\hat{q})\right)^T$. This equality certainly does not hold for all multiscale densities q because, at the very least, the transformed density $\bar{q} \triangleq \left(\mathcal{F}^M(q)\right)^T$ would have to satisfy $\bar{q}(x_v) = p^*(x_v) = q(x_v)$ for all $v \in M$. Therefore, solutions to problem $\widetilde{\mathcal{Q}}$ have a remarkable invariance property, as evidenced by the following proposition.

Proposition 3.21 (An Invariance Property of Solutions to Problem $\widetilde{\mathcal{Q}}$).
If $\hat{q}$ is a solution to problem $\widetilde{\mathcal{Q}}$, then $\hat{q} = \left(\mathcal{F}^M(\hat{q})\right)^T$.

Proof. To prove this result, we consider what happens when the mappings $\mathcal{F}^M(\cdot)$ and $p \longrightarrow p^T$ are successively applied to $\hat{q}$. In particular, we define the densities $\hat{p} \triangleq \mathcal{F}^M(\hat{q})$ and $\bar{p} \triangleq \mathcal{F}^M(\hat{p}^T)$, or equivalently, we consider the following succession of densities,

$$\hat{q} \xrightarrow{\mathcal{F}^M(\cdot)} \hat{p} \xrightarrow{(\cdot)^T} \hat{p}^T \xrightarrow{\mathcal{F}^M(\cdot)} \bar{p}.$$

Using Proposition 3.19, the divergence $D(\hat{p}\|\hat{q})$ may be decomposed as follows,

$$D(\hat{p}\|\hat{q}) = D\left(\hat{p}\|\hat{p}^T\right) + D\left(\hat{p}^T\|\hat{q}\right), \tag{3.60}$$

and since $\hat{p} \in \mathcal{P}^M(V, d)$ and $\hat{p}^T \in \mathcal{P}(V, d)$, the first term on the right-hand side of (3.60) may be further decomposed using the result in Proposition 3.20,

$$D(\hat{p}\|\hat{q}) = D(\hat{p}\|\bar{p}) + D\left(\bar{p}\|\hat{p}^T\right) + D\left(\hat{p}^T\|\hat{q}\right). \tag{3.61}$$

Using the relationship in (3.57), we can also write $D(\hat{p}\|\hat{q}) = D(p^*(x_M)\|\hat{q}(x_M))$ and $D\left(\bar{p}\|\hat{p}^T\right) = D\left(p^*(x_M)\|\hat{p}^T(x_M)\right)$, so that (3.61) is equivalent to

$$D(p^*(x_M)\|\hat{q}(x_M)) = D\left(p^*(x_M)\|\hat{p}^T(x_M)\right) + D(\hat{p}\|\bar{p}) + D\left(\hat{p}^T\|\hat{q}\right). \tag{3.62}$$

By the non-negativity of the KL divergence, $D(p^*(x_M)\|\hat{q}(x_M)) \geq D\left(p^*(x_M)\|\hat{p}^T(x_M)\right)$ with equality if and only if $\hat{p} = \bar{p}$ and $\hat{p}^T = \hat{q}$. However, if $D(p^*(x_M)\|\hat{q}(x_M)) > D\left(p^*(x_M)\|\hat{p}^T(x_M)\right)$ then this contradicts the fact that $\hat{q}$ is a solution to problem $\widetilde{\mathcal{Q}}$. Thus, we must have equality, in which case $\hat{p}^T = \hat{q}$ which then implies $\hat{p} = \bar{p}$. ■

This invariance property therefore requires all solutions $\hat{q}$ of problem $\widetilde{\mathcal{Q}}$ to satisfy $\hat{q}(x_v) = p^*(x_v)$ for all $v \in M$, or in other words, all solutions to problem $\widetilde{\mathcal{Q}}$ must, at the very least, match the vertex marginals $p^*(x_v)$ of the target density.

From a slightly different perspective, this invariance property is equivalent to the statement that $p \longrightarrow p^T$ is an inverse mapping of $\mathcal{F}^M(\cdot)$ on the domain equal to the solution set of problem $\widetilde{\mathcal{Q}}$. Recall from Section 3.4 that we proved the identity map to be a right inverse of the mapping $p \longrightarrow p^T$ when applied to solutions of the exact realization problem. As we show in the next section, the mapping $\mathcal{F}^M(\cdot)$ is an appropriate generalization of the identity mapping to the approximate realization problem, and it is used to prove that $p \longrightarrow p^T$ is a surjection for a number of alternative realization problems.

■ 3.10.3 Necessary Conditions for Approximate Multiscale Realization

In this section, we provide a generalization of the ideas discussed in Section 3.4. Namely, we suggest a series of alternative formulations to problem $\widetilde{\mathcal{Q}}$, and we show that a subset of these alternatives is equivalent to $\widetilde{\mathcal{Q}}$ in the sense that a surjective mapping exists from the solution set of each alternative problem to the solution set of problem $\widetilde{\mathcal{Q}}$. This analysis also suggests necessary conditions for the approximate realization problem.

Alternative Problem Formulations for Approximate Multiscale Realization

Rather than solving problem $\widetilde{\mathcal{Q}}$ directly, consider solving one of a series of alternative problems which searches over a set of densities different from the multiscale densities considered in problem $\widetilde{\mathcal{Q}}$; specifically, the search is performed over the set $\mathcal{P}_{\mathcal{G}}^M(V, d)$ defined in (3.22) for some specified graph $\mathcal{G}$. Therefore, the problems considered here are similar to the alternative problems $\mathcal{P}_{\mathcal{G}}^M$ discussed in Section 3.4.2 in that each density $p \in \mathcal{P}_{\mathcal{G}}^M(V, d)$ possesses the conditional independence structure implied by $\mathcal{G}$ and has the marginal $p(x_M) = p^*(x_M)$. However, the problems considered here are relaxed in the sense that the marginal $p^T(x_M)$ is not required to match $p^*(x_M)$ exactly but rather minimize the functional $D\left(p^*(x_M) \| p^T(x_M)\right)$.

Given a rooted tree $\mathcal{G}_{\preceq}$ defined on a vertex set V and any graph $\mathcal{G} = (V, E)$, consider the following approximation to alternative problem $\mathcal{P}_{\mathcal{G}}^M$, henceforth called *alternative approximate problem* $\widetilde{\mathcal{P}}_{\mathcal{G}}^M$.

<u>Alternative Approximate Problem $\widetilde{\mathcal{P}}_{\mathcal{G}}^M$</u>: Find any density $\hat{p} \in \mathcal{P}_{\mathcal{G}}^M(V, d)$ which minimizes the cost $D\left(p^*(x_M) \| \hat{p}^T(x_M)\right)$, *i.e.*

$$\hat{p} = \arg \min_{p \in \mathcal{P}_{\mathcal{G}}^M(V,d)} D\left(p^*(x_M) \| p^T(x_M)\right).$$

Therefore, problem $\widetilde{\mathcal{P}}_{\mathcal{G}}^M$ seems to be a relaxation of problem $\mathcal{P}_{\mathcal{G}}^M$ in the sense that $\hat{p}^T(x_M)$ is not required to be equal to $p^*(x_M)$. In the special case where $\mathcal{G}$ is the complete graph, we often use the notation $\widetilde{\mathcal{P}}^M$ rather that $\widetilde{\mathcal{P}}_{\mathcal{G}}^M$ to emphasize the fact that $\widetilde{\mathcal{P}}^M$ is an approximation to problem $\mathcal{P}^M$ introduced in Section 3.4.1.

For the exact realization problem, Proposition 3.3 proved the mapping $p \longrightarrow p^T$ to be a surjection from the solution set of $\mathcal{P}_{\mathcal{G}}^M$ onto the solution set of $\mathcal{Q}$, as long as $\mathcal{G}$ is a supergraph of $\mathcal{G}_{\preceq}^{\sim}$.

In the approximate realization problem, this same result does not necessarily hold; specifically, a surjection may not exist from the solution set of problem $\widetilde{\mathcal{P}}_{\mathcal{G}}^{M}$ onto the solution set of problem $\widetilde{\mathcal{Q}}$, even if $\mathcal{G}$ is a supergraph of $\mathcal{G}_{\precsim}$. This is due to the fact that problem $\widetilde{\mathcal{Q}}$ is by definition a relaxation of problem $\mathcal{Q}$, while problem $\widetilde{\mathcal{P}}_{\mathcal{G}}^{M}$ may or may not be a relaxation of problem $\mathcal{P}_{\mathcal{G}}^{M}$ for some choices of $\mathcal{G}$. The following example illustrates this idea.

Example 3.6 (On the Existence of Surjective Mappings).
This example provides a thought exercise to illustrate the fact that $\widetilde{\mathcal{P}}_{\mathcal{G}}^{M}$ may not be a relaxation of problem $\mathcal{P}_{\mathcal{G}}^{M}$ for all graphs $\mathcal{G}$ and to show how this affects the existence of a surjection between the solution sets of $\widetilde{\mathcal{P}}_{\mathcal{G}}^{M}$ and $\widetilde{\mathcal{Q}}$. Consider again the graphs shown in Figure 3.6, and suppose a target density $p^*(x_M)$ is given for $M = \{1, 2, 3\}$, the leaf vertices of the rooted tree shown in Figure 3.6(a). For the purpose of this example, we assume that problem $\widetilde{\mathcal{Q}}$ has at least one solution, while problem $\mathcal{Q}$ has no solution. This is certainly possible since $\widetilde{\mathcal{Q}}$ is a relaxation of $\mathcal{Q}$.

Consider now the graph $\mathcal{G}$ shown in Figure 3.6(c), where $\mathcal{G}$ is a supergraph of $\mathcal{G}_{\precsim}$. For some choice of $p^*(x_M)$ and a sufficiently small choice for the dimension of random vector X_0, it is possible that $\mathcal{P}_{\mathcal{G}}^{M}(V, d)$ is empty because there is no density p which has dimensions d, factors according to $\mathcal{G}$, and exactly satisfies the marginal constraint $p(x_M) = p^*(x_M)$. Then, if $\mathcal{P}_{\mathcal{G}}^{M}(V, d)$ is empty, neither problem $\mathcal{P}_{\mathcal{G}}^{M}$ nor problem $\widetilde{\mathcal{P}}_{\mathcal{G}}^{M}$ has a solution since each searches over the space $\mathcal{P}_{\mathcal{G}}^{M}(V, d)$, and since problem $\widetilde{\mathcal{Q}}$ does have a solution, there is no surjection from the solution set of $\widetilde{\mathcal{P}}_{\mathcal{G}}^{M}$ onto the solution set of $\widetilde{\mathcal{Q}}$.

Notice that the graph $\mathcal{G}$ shown in Figure 2.8(c) does not have a clique equal to $M = \{1, 2, 3\}$. Because of this, we cannot guarantee that the set $\mathcal{P}_{\mathcal{G}}^{M}(V, d)$ is non-empty, and as a result, the search space for problem $\widetilde{\mathcal{P}}_{\mathcal{G}}^{M}$ may be overly restrictive. In order to ensure that $\mathcal{P}_{\mathcal{G}}^{M}(V, d)$ is non-empty, the graph $\mathcal{G}$ must satisfy the following two properties:

(1) $\mathcal{G}$ has a clique equal to M,

(2) $\mathcal{G}$ is a supergraph of $\mathcal{G}_{\precsim}$.

As discussed in the previous section, the properties of the mapping $\mathcal{F}^M(\cdot)$ ensure that the set $\mathcal{P}_{\mathcal{G}}^{M}(V, d)$ is non-empty when $\mathcal{G}$ satisfies these two conditions. Because of this, we choose to focus on these types of graphs in the remainder of our discussion, and in particular, we prove that the approximate problem $\widetilde{\mathcal{P}}_{\mathcal{G}}^{M}$ is, in this case, a sufficient relaxation of $\mathcal{P}_{\mathcal{G}}^{M}$ to allow a surjection between the solution sets of $\widetilde{\mathcal{P}}_{\mathcal{G}}^{M}$ and $\widetilde{\mathcal{Q}}$. ◀

Consider now any alternative problem $\widetilde{\mathcal{P}}_{\mathcal{G}}^{M}$ where $\mathcal{G}$ is a supergraph of $\mathcal{G}_{\precsim}$ with a clique equal to M. In the following proposition, we prove that $p \longrightarrow p^T$ is a surjection from the solution set of $\widetilde{\mathcal{P}}_{\mathcal{G}}^{M}$ onto the solution set of problem $\widetilde{\mathcal{Q}}$. To do this, we first show that $p \longrightarrow p^T$ is a mapping from the solution set of $\widetilde{\mathcal{P}}_{\mathcal{G}}^{M}$ to the solution set of $\widetilde{\mathcal{Q}}$, and second, we show that the mapping $\mathcal{F}^M(\cdot)$ is a right inverse of $p \longrightarrow p^T$. These two facts prove that $p \longrightarrow p^T$ is in fact a surjection.

Proposition 3.22 (Relationship Between Solutions to $\widetilde{\mathcal{P}}_{\mathcal{G}}^{M}$ and $\widetilde{\mathcal{Q}}$).
Let $\mathcal{G}_{\preceq}$ be a rooted tree defined on vertex set V, and let $p^*(x_M)$ be a given target density. If a graph $\mathcal{G} = (V, E)$ is a supergraph of $\mathcal{G}_{\precsim}$ and has a clique equal to M, the mapping $p \longrightarrow p^T$ is a surjection from the solution set of problem $\widetilde{\mathcal{P}}_{\mathcal{G}}^{M}$ onto the solution set of problem $\widetilde{\mathcal{Q}}$.

Proof. See Appendix B.11. ■

As a consequence of Proposition 3.22, it is reasonable to consider solving the alternative approximate problem $\widetilde{\mathcal{P}}_G^M$ rather than problem $\widetilde{\mathcal{Q}}$ since all solutions to problem $\widetilde{\mathcal{Q}}$ may be identified from solutions to problem $\widetilde{\mathcal{P}}_G^M$.

Necessary Conditions for Alternative Problem Formulations

Recall from Section 3.4.3 that the following two constraints are sufficient conditions for solutions to alternative problem $\mathcal{P}_G^M$ and hence the exact realization problem:

$$\underline{\text{Condition 1}} \quad \hat{p} \in \mathcal{P}_G^M(V, d),$$

$$\underline{\text{Condition 2}} \quad \hat{p} = \hat{p}^T.$$

Compare these two constraints with the following, which we subsequently show in Theorem 3.8 to be necessary conditions for solutions to alternative approximate problem $\widetilde{\mathcal{P}}_G^M$:

$$\underline{\text{Condition 1}} \quad \hat{p} \in \mathcal{P}_G^M(V, d),$$

$$\underline{\text{Condition 2}} \quad \hat{p}^T = \left[\mathcal{F}^M(\hat{p}^T)\right]^T.$$

The first constraint is the same in both sets of conditions and simply requires $\hat{p}$ to lie in the search set for problems $\mathcal{P}_G^M$ and $\widetilde{\mathcal{P}}_G^M$. The second constraint in the latter set of conditions is more interesting because it is (as it should be) less-stringent than the second constraint in the former set of conditions. In particular, if there is no exact solution to the realization problem, then the constraints $\hat{p} \in \mathcal{P}_G^M(V, d)$ and $\hat{p} = \hat{p}^T$ cannot be simultaneously satisfied by any density $\hat{p}$, and in this case, the necessary condition $\hat{p}^T = \left[\mathcal{F}^M(\hat{p}^T)\right]^T$ represents a relaxation of the constraint $\hat{p} = \hat{p}^T$.

On the other hand, the latter set of constraints can be extremely weak for some realization problems. In particular, the constraint $\hat{p}^T = \left[\mathcal{F}^M(\hat{p}^T)\right]^T$ is an uninformative necessary condition for solutions to exact realization problem $\mathcal{P}_G^M$. To see this, recall that every solution $\hat{p}$ to problem $\mathcal{P}_G^M$ satisfies $\hat{p}^T(x_M) = p^*(x_M)$, and consequently, the density $\hat{p}^T$ remains unchanged under the mapping $\mathcal{F}^M(\cdot)$, *i.e.* $\hat{p}^T = \mathcal{F}^M(\hat{p}^T)$. Using this fact, the constraint $\hat{p}^T = \left[\mathcal{F}^M(\hat{p}^T)\right]^T$ simply requires $\hat{p}^T = \left[\hat{p}^T\right]^T$. However, every density satisfies this constraint due to the fact that $(\cdot)^T$ is a projection operator, and therefore, the latter set of constraints are uninformative necessary conditions for the exact realization problem.

Figure 3.22 provides a graphical illustration of these two sets of conditions. The illustration provided in Figure 3.22(a) shows the case where the realization problem has an exact solution. In this situation, the sufficient conditions $\hat{p} \in \mathcal{P}_G^M(V, d)$ and $\hat{p} = \hat{p}^T$ precisely characterize the densities which lie in the intersection of the sets $\mathcal{P}_G^M(V, d)$ and $\mathcal{P}_{G_{\preceq}}(V, d)$, *i.e.* multiscale densities which exactly match the target density $p^*(x_M)$. The illustration provided in Figure 3.22(b) shows the case where the realization problem has no exact solution. In this situation, the necessary conditions for approximate problem $\widetilde{\mathcal{P}}_G^M$ (the latter set of conditions) require every solution $\hat{p}$ to satisfy $\hat{p} \in \mathcal{P}_G^M(V, d)$ and $\hat{p}^T = \bar{p}^T$ with $\bar{p} \triangleq \mathcal{F}^M(\hat{p}^T)$. As the figure illustrates, when the two sets $\mathcal{P}_G^M(V, d)$ and $\mathcal{P}_{G_{\preceq}}(V, d)$ do not intersect, the latter set of constraints characterizes solutions $\hat{p}$ in terms of their orthogonality properties.

Figure 3.22(b) also illustrates the fact that some solutions to problem $\widetilde{\mathcal{P}}_G^M$ satisfy an interesting invariance property. Namely, there exists a solution $\bar{p} \in \mathcal{P}_G^M(V, d)$ which remains invariant under the composition of the two projections $(\cdot)^T$ and $\mathcal{F}^M(\cdot)$, and furthermore, every solution $\hat{p}$ to problem $\widetilde{\mathcal{P}}_{\mathcal{G}}^M$ can be mapped to such a density as shown in Figure 3.22(b). We exploit this property in the next chapter in developing an iterative approach for solving the approximate realization problem.

Using the invariance property of the mapping $\mathcal{F}^M(\cdot)$ provided in Proposition 3.21, along with Proposition 3.22, it is straightforward to show that the latter set of constraints are indeed necessary conditions. This result is formally stated and proven in the following theorem.

Theorem 3.8 (Necessary Conditions for Problem $\widetilde{\mathcal{P}}_{\mathcal{G}}^M$).
Let $\mathcal{G}_{\preceq}$ be a rooted tree defined on vertex set V, and let $p^*(x_M)$ be a given target density. Suppose $\mathcal{G} = (V, E)$ is any triangulated supergraph of $\mathcal{G}_{\precsim}$ with a clique equal to M. If density $\hat{p}$ is a solution to problem $\widetilde{\mathcal{P}}_{\mathcal{G}}^M$, then it must satisfy $\hat{p} \in \mathcal{P}_{\mathcal{G}}^M(V, d)$ and $\hat{p}^T = \left[\mathcal{F}^M(\hat{p}^T)\right]^T$.

Proof. The constraint $\hat{p} \in \mathcal{P}_{\mathcal{G}}^M(V, d)$ requires $\hat{p}$ to lie in the search set for problem $\widetilde{\mathcal{P}}_{\mathcal{G}}^M$. Using Proposition 3.22, we know that $\hat{q} \triangleq \hat{p}^T$ is a solution to problem $\widetilde{\mathcal{Q}}$, and then, by Proposition 3.21, we know that this solution must satisfy $\hat{q} = \left[\mathcal{F}^M(\hat{q})\right]^T$, thereby proving the result. ■

In the preceding discussion, we have focused exclusively on graphs $\mathcal{G}$ which are supergraphs of a tree $\mathcal{H} = \mathcal{G}_{\precsim}$. However, all of the results provided for the approximate multiscale realization generalize to the case where $\mathcal{H}$ is any triangulated graph, not necessarily a tree. Consequently, by using the notion of an augmented graph, a result similar to Theorem 3.8 also holds for the state augmentation problem. While we do not derive this fact here, the necessary conditions for the approximate realization problem with augmented states are identical to those provided in Theorem 3.8 if the assumptions in Theorem 3.4 are used.

■ 3.10.4 An Important Decomposition of the Kullback-Leibler Divergence

Recall that Proposition 3.20 provides an important decomposition of the KL divergence which emphasizes the geometry of the mapping $\mathcal{F}^M(\cdot)$. The next chapter focuses on this result in further detail and shows how the relationship in (3.58) leads to an iterative algorithm for solving the approximate realization problem. In concluding this chapter, we show how the decomposition in (3.58) is related to both the sufficient conditions for exact realization and the necessary conditions for approximate realization.

Choose any $p \in \mathcal{P}^M(V, d)$, and notice that $p_{\mathcal{G}} \in \mathcal{P}^M(V, d)$ as long as $\mathcal{G}$ is a triangulated graph with a clique equal to M. Then, applying the relationship in (3.58) gives

$$D\left(p_{\mathcal{G}}(x) \| p^T(x)\right) = D\left(p_{\mathcal{G}}(x) \| \mathcal{F}^M(p^T(x))\right) + D\left(\mathcal{F}^M(p^T(x)) \| p^T(x)\right),$$

and since the final term may be rewritten using (3.57), we get the following important decomposition,

$$D\left(p^*(x_M) \| p^T(x_M)\right) = D\left(p_{\mathcal{G}}(x) \| p^T(x)\right) - D\left(p_{\mathcal{G}}(x) \| \mathcal{F}^M(p^T(x))\right). \tag{3.63}$$

This relationship is intuitively pleasing because it indicates that the "distance" between $p^*(x_M)$ and $p^T(x_M)$ may equivalently be measured by computing the "total distance" between $p_{\mathcal{G}}$ and p^T and subtracting the "distance" between $p_{\mathcal{G}}$ and $\mathcal{F}^M(p^T)$. Furthermore, the density $p_{\mathcal{G}}$ in (3.63) may be replaced by any projection $p_{\mathcal{G}'}$, as long as $\mathcal{G}'$ is a triangulated graph with a clique equal to M. These ideas are graphically depicted in Figure 3.23.

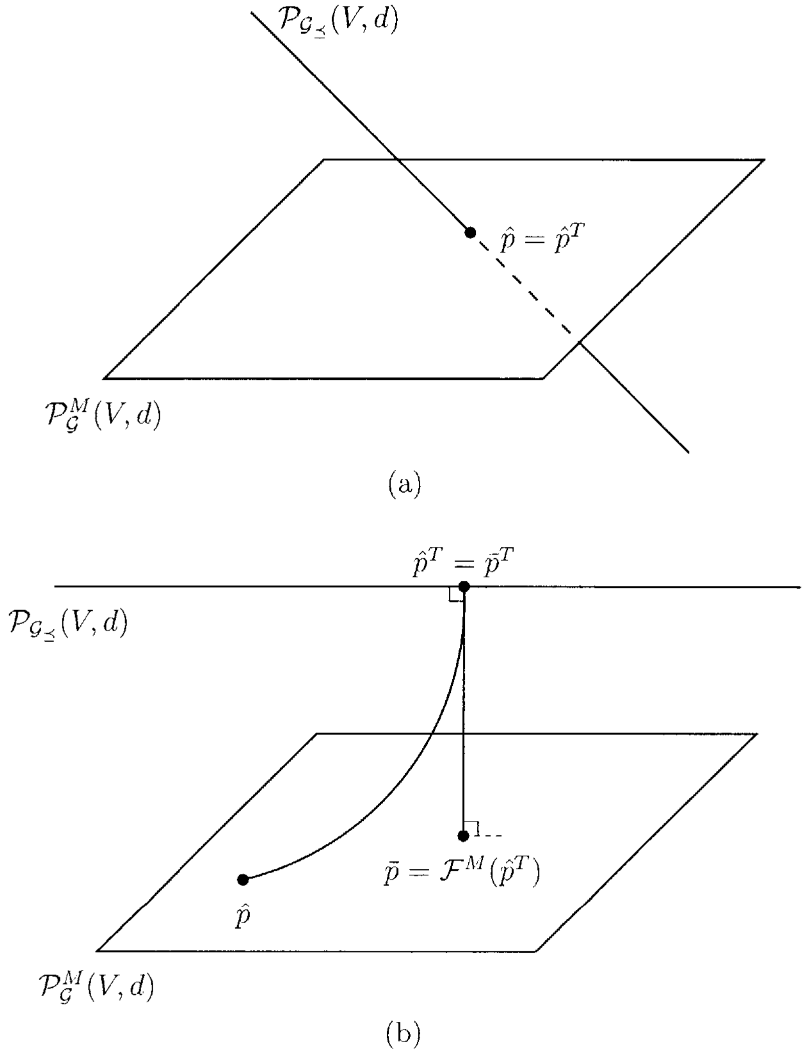

$\mathcal{P}_{\mathcal{G}_\preceq}(V,d)$
$\hat{p} = \hat{p}^T$
$\mathcal{P}_{\mathcal{G}}^M(V,d)$
(a)
$\hat{p}^T = \bar{p}^T$
$\mathcal{P}_{\mathcal{G}_\preceq}(V,d)$
$\bar{p} = \mathcal{F}^M(\hat{p}^T)$
$\hat{p}$
$\mathcal{P}_{\mathcal{G}}^M(V,d)$
(b)

Figure 3.22. (a) Graphical depiction of sufficient conditions for exact realization problem $\mathcal{P}_{\mathcal{G}}^M$. (b) Graphical depiction of necessary conditions for approximate realization problem $\widetilde{\mathcal{P}}_{\mathcal{G}}^M$.

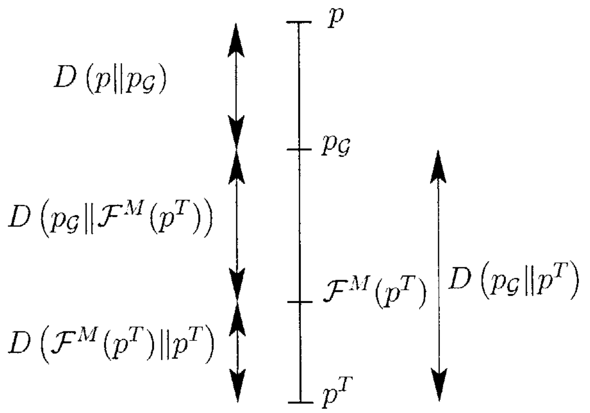

p
$D(p\|p_{\mathcal{G}})$
$p_{\mathcal{G}}$
$D(p_{\mathcal{G}}\|\mathcal{F}^M(p^T))$
$\mathcal{F}^M(p^T)$
$D(p_{\mathcal{G}}\|p^T)$
$D(\mathcal{F}^M(p^T)\|p^T)$
p^T

Figure 3.23. Illustrates the additive property of the Kullback-Leibler divergence for the two types of projections $p_{\mathcal{G}}$ and $\mathcal{F}^M(p^T)$.

The Tradeoff Between Sufficient Conditions for Exact Realization and Necessary Conditions for Approximate Realization

Besides the preceding additive property, there are several important points to make about the decomposition in (3.63) in terms of the sufficient conditions for exact realization and the necessary conditions for approximate realization:

(1) Due to the non-negativity of the KL divergence, (3.63) shows that $D\left(p_{\mathcal{G}}(x)\|p^T(x)\right)$ is an upper bound for $D\left(p^*(x_M)\|p^T(x_M)\right)$. Consequently, if there exists a density $p \in \mathcal{P}^M(V,d)$ such that $p_{\mathcal{G}} = p^T$, then p^T is an exact solution to the realization problem. These same sufficient conditions were previously derived in Theorem 3.3 for the exact realization problem.

(2) Consider now the second term on the right-hand-side of (3.63) which measures the divergence between $p_{\mathcal{G}}$ and $\mathcal{F}^M(p^T)$. Notice that any density p which satisfies $p_{\mathcal{G}} = \mathcal{F}^M(p^T)$ must also satisfy $p^T = \left[\mathcal{F}^M(p^T)\right]^T$, and consequently, the term $D\left(p_{\mathcal{G}}\|\mathcal{F}^M(p^T)\right)$ measures something stronger than the necessary conditions stated in Theorem 3.8. However, it is important to note that the conditions $p \in \mathcal{P}^M(V,d)$ and $p_{\mathcal{G}} = \mathcal{F}^M(p^T)$ are neither necessary nor sufficient for the approximate realization problem. As Lemma 3.9 (to follow) shows, though, there exists at least one solution to the approximate realization problem which satisfies these two conditions.

(3) As a whole, the relationship in (3.63) shows a tradeoff between the sufficient conditions for exact realization and the necessary conditions for approximate realization. Specifically, if we try to minimize the upper bound $D\left(p_{\mathcal{G}}(x)\|p^T(x)\right)$ or equivalently seek to satisfy the sufficient conditions for the exact realization problem, the actual divergence $D\left(p^*(x_M)\|p^T(x_M)\right)$ is smaller than this upper bound because the sufficient conditions for exact realization are too stringent for the approximate realization problem. Consequently, the term $D\left(p_{\mathcal{G}}\|\mathcal{F}^M(p^T)\right)$ provides the appropriate correction factor. As Proposition 3.23 (to follow) states, if for some density $p \in \mathcal{P}^M(V,d)$ the upper bound $D\left(p_{\mathcal{G}}\|p^T\right)$ cannot be further decreased, then p^T is a solution to approximate realization problem $\tilde{\mathcal{Q}}$, and furthermore, $D\left(p_{\mathcal{G}}\|\mathcal{F}^M(p^T)\right) = 0$ so that the condition $p_{\mathcal{G}} = \mathcal{F}^M(p^T)$ (and in turn the necessary condition $p^T = \left[\mathcal{F}^M(p^T)\right]^T$) is

exactly satisfied in this case.

As the second item in the preceding discussion suggests, there always exists a solution to alternative problem $\widetilde{\mathcal{P}}^M$ which satisfies the constraint $\hat{p}_{\mathcal{G}} = \mathcal{F}^M(\hat{p}^T)$. For example, the density $\bar{p}$ illustrated in Figure 3.22(b) is one such solution which satisfies this constraint. The following lemma states this important result.

Lemma 3.9 (Existence of Solutions Satisfying $\hat{p}_{\mathcal{G}} = \mathcal{F}^M(\hat{p}^T)$).
Let $\mathcal{G}$ be a graph satisfying the assumptions stated in Theorem 3.8. If approximate realization problem $\widetilde{\mathcal{Q}}$ has at least one solution, then there exists a solution $\hat{p}$ to problem $\widetilde{\mathcal{P}}^M$ satisfying the conditions $\hat{p} \in \mathcal{P}^M(V,d)$ and $\hat{p}_{\mathcal{G}} = \mathcal{F}^M(\hat{p}^T)$.

Proof. Suppose problem $\widetilde{\mathcal{Q}}$ has at least one solution $\hat{q}$. The proof to Proposition 3.22 shows that $\hat{p} \triangleq \mathcal{F}^M(\hat{q})$ is a solution to problem $\widetilde{\mathcal{P}}^M$, and consequently, the solution set of problem $\widetilde{\mathcal{P}}^M$ is non-empty.

Now, choose any solution $\hat{p}$ to problem $\widetilde{\mathcal{P}}^M$, and consider the density $\bar{p} \triangleq \mathcal{F}^M(\hat{p}^T)$. In most cases, the two densities $\hat{p}$ and $\bar{p}$ are not the same, but using Proposition 3.22, we know that $\bar{p}$ is a solution to problem $\widetilde{\mathcal{P}}^M$. Furthermore, using the invariance property in Proposition 3.21, the densities $\hat{p}$ and $\bar{p}$ satisfy $\hat{p}^T = \left(\mathcal{F}^M(\hat{p}^T)\right)^T = \bar{p}^T$. Therefore, the density $\bar{p}$ satisfies $\bar{p} \in \mathcal{P}^M(V,d)$ and $\bar{p} = \mathcal{F}^M(\bar{p}^T)$. Finally, by the discussion in Section 3.10.2, $\bar{p} = \mathcal{F}^M(\bar{p}^T)$ must factor according to all supergraphs of $\mathcal{G}_{\precsim}$ which have a clique equal to M. Consequently, $\bar{p} = \bar{p}_{\mathcal{G}}$ for the graphs $\mathcal{G}$ considered in Theorem 3.8, thereby proving the result. ■

Using Lemma 3.9, we can also prove an important result about the upper bound $D\left(p_{\mathcal{G}}(x) \| p^T(x)\right)$ in (3.63). Specifically, as mentioned in the preceding discussion, minimizing this upper bound leads to a solution to the approximate realization problem, as stated in the following proposition.

Proposition 3.23 (Identifying Solutions By Minimizing An Upper Bound).
Let $\mathcal{G}$ be a graph satisfying the assumptions stated in Theorem 3.8. A density $\hat{p}$ minimizes $D\left(p_{\mathcal{G}}(x) \| p^T(x)\right)$ over all $p \in \mathcal{P}^M(V,d)$ if and only if $\hat{p}$ is both a solution to problem $\widetilde{\mathcal{P}}^M$ and satisfies $\hat{p}_{\mathcal{G}} = \mathcal{F}^M(\hat{p}^T)$.

Proof. Let $\hat{p}$ be a minimizer of $D\left(p_{\mathcal{G}}(x) \| p^T(x)\right)$ over all $p \in \mathcal{P}^M(V,d)$. Since $\mathcal{G}$ has a clique equal to M, $\hat{p}_{\mathcal{G}} \in \mathcal{P}^M_{\mathcal{G}}(V,d)$, and consequently, we can use the relationship in (3.63) to write the following,

$$D\left(p^*(x_M) \| \hat{p}^T(x_M)\right) = D\left(\hat{p}_{\mathcal{G}}(x) \| \hat{p}^T(x)\right) - D\left(\hat{p}_{\mathcal{G}}(x) \| \mathcal{F}^M(\hat{p}^T(x))\right). \tag{3.64}$$

Now, suppose $\hat{p}$ is not a solution to problem $\widetilde{\mathcal{P}}^M$, so that there exists a density p which satisfies $D\left(p^*(x_M) \| p^T(x_M)\right) < D\left(p^*(x_M) \| \hat{p}^T(x_M)\right)$. By Lemma 3.9, there exists a solution p which satisfies $p_{\mathcal{G}} = \mathcal{F}^M(p^T)$, and using (3.63), this implies $D\left(p^*(x_M) \| p^T(x_M)\right) = D\left(p_{\mathcal{G}}(x) \| p^T(x)\right)$. Combining this fact with (3.64) gives the following,

$$D\left(p^*(x_M) \| p^T(x_M)\right) = D\left(p_{\mathcal{G}}(x) \| p^T(x)\right) < D\left(\hat{p}_{\mathcal{G}}(x) \| \hat{p}^T(x)\right) - D\left(\hat{p}_{\mathcal{G}}(x) \| \mathcal{F}^M(\hat{p}^T(x))\right),$$

which indicates that $D\left(p_{\mathcal{G}}(x) \| p^T(x)\right) < D\left(\hat{p}_{\mathcal{G}}(x) \| \hat{p}^T(x)\right)$. However, this contradicts the minimality of $\hat{p}$. Consequently, there is no such density p, and $\hat{p}$ is a solution to problem $\widetilde{\mathcal{P}}^M$. Furthermore, this same argument implies that $D\left(\hat{p}_{\mathcal{G}} \| \mathcal{F}^M(\hat{p}^T)\right) = 0$ so that $\hat{p}_{\mathcal{G}} = \mathcal{F}^M(\hat{p}^T)$.

Now suppose $\hat{p}$ is both a solution to problem $\widetilde{\mathcal{P}}^M$ and satisfies $\hat{p}_{\mathcal{G}} = \mathcal{F}^M(\hat{p}^T)$. Suppose however that $\hat{p}$ is not a minimizer of $D\left(p_{\mathcal{G}}(x) \| p^T(x)\right)$, so that there exists a density p satisfying $D\left(p_{\mathcal{G}}(x) \| p^T(x)\right) < D\left(\hat{p}_{\mathcal{G}}(x) \| \hat{p}^T(x)\right)$. Using the relationship in (3.63) on both sides of the preceding equation, along with the fact that $\hat{p}_{\mathcal{G}} = \mathcal{F}^M(\hat{p}^T)$, gives the following,

$$D\left(p^*(x_M) \| p^T(x_M)\right) + D\left(p_{\mathcal{G}}(x) \| \mathcal{F}^M\left(p^T(x)\right)\right) < D\left(p^*(x_M) \| \hat{p}^T(x_M)\right).$$

This implies that $D\left(p^*(x_M) \| p^T(x_M)\right) < D\left(p^*(x_M) \| \hat{p}^T(x_M)\right)$, which contradicts the fact that $\hat{p}$ is a solution to problem $\widetilde{\mathcal{P}}^M$. Hence, there is no such density p, and $\hat{p}$ is a minimizer of $D\left(p_{\mathcal{G}}(x) \| p^T(x)\right)$ over all $p \in \mathcal{P}^M(V, d)$. ■

A Greedy Approach to Approximate Realization

The result provided in Proposition 3.23 suggests one approach to finding a solution to the approximate realization problem. Specifically, rather than minimizing $D\left(p^*(x_M) \| p^T(x_M)\right)$, we can instead attempt to minimize the upper bound $D\left(p_{\mathcal{G}}(x) \| p^T(x)\right)$ over all densities $p \in \mathcal{P}^M(V, d)$. Of course, this approach to the realization problem does not represent a significant simplification since minimizing the upper bound is still a global minimization problem. As we show here, though, the ideas previously discussed within the context of the exact realization problem may be used to decompose this global cost function into a sum of more localized cost functions.

In particular, we propose an expansion of $D\left(p_{\mathcal{G}} \| p^T\right)$ into a sum of terms, based on a chosen sequence of clique extensions $\mathcal{G}_0 = \mathcal{G}_{\precsim}, \mathcal{G}_1, \ldots, \mathcal{G}_n = \mathcal{G}$. Based on the discussion in Section 3.7.1 and Lemma 3.6 in particular, such a sequence must exist, since $\mathcal{G}_{\precsim}$ is triangulated and since $\mathcal{G}$ is a triangulated supergraph of $\mathcal{G}_{\precsim}$. Given such a sequence, the corresponding projections $p_{\mathcal{G}_0}, \ldots, p_{\mathcal{G}_n}$ satisfy the simple probabilistic relationship in (3.39) for clique extensions, and using this fact in conjunction with the additivity properties (3.54) and (3.55) of KL proves that $D\left(p_{\mathcal{G}} \| p^T\right)$ may be decomposed into a sum of simpler terms. The following proposition provides such a decomposition for any sequence of clique extensions between two triangulated graphs $\mathcal{G}$ and $\mathcal{G}'$.

Proposition 3.24 (Decomposing KL for a Sequence of Clique Extensions).
Let $\mathcal{G} = (V, E)$ be a triangulated graph, and let $\mathcal{G} = \mathcal{G}_0, \mathcal{G}_1, \mathcal{G}_2, \ldots, \mathcal{G}_n = \mathcal{G}'$ be a sequence of graphs where $\mathcal{G}_i$ is a clique extension of $\mathcal{G}_{i-1}$ and where C_i is the unique maximal clique contained in $\mathcal{G}_i$ but not $\mathcal{G}_{i-1}$. Then, for any density $p(x_V)$, the following decomposition of the KL divergence holds,

$$D\left(p_{\mathcal{G}'}(x) \| p_{\mathcal{G}}(x)\right) = \sum_{i=1}^{n} D\left(p(x_{C_i}) \| p_{\mathcal{G}_{i-1}(C_i)}(x_{C_i})\right). \tag{3.65}$$

Proof. This result can be derived directly using the relationship in Corollary 3.2 and the definition of KL; we instead choose to derive the result using the properties of KL. First, suppose $n = 1$ so that $\mathcal{G}'$ is a clique extension of $\mathcal{G}$ with unique new maximal clique C. Then, using (3.54), we get

$$D\left(p_{\mathcal{G}'}(x) \| p_{\mathcal{G}}(x)\right) = D\left(p_{\mathcal{G}'}(x_C) \| p_{\mathcal{G}}(x_C)\right) + D\left(p_{\mathcal{G}'}(x_{V-C} | x_C) \| p_{\mathcal{G}}(x_{V-C} | x_C)\right). \tag{3.66}$$

By the relationship in (3.39) for clique extensions, the last term in (3.66) is zero because the two conditional densities are identical. Therefore, we get

$$D\left(p_{\mathcal{G}'}(x) \| p_{\mathcal{G}}(x)\right) = D\left(p(x_C) \| p_{\mathcal{G}}(x_C)\right) = D\left(p(x_C) \| p_{\mathcal{G}(C)}(x_C)\right). \tag{3.67}$$

Suppose now that $n > 1$. By the nature of the clique extension, each $\mathcal{G}_i$ for $i = 1, \ldots, n$ is a triangulated supergraph of $\mathcal{G}_0$. As such, we can inductively use the additivity property (3.55) of projections to show the following,

$$D\left(p_{\mathcal{G}'}(x) \| p_{\mathcal{G}}(x)\right) = D\left(p_{\mathcal{G}_n}(x) \| p_{\mathcal{G}_0}(x)\right) = \sum_{i=1}^{n} D\left(p_{\mathcal{G}_i}(x) \| p_{\mathcal{G}_{i-1}}(x)\right). \tag{3.68}$$

Combining (3.67) and (3.68) proves the result. ■

As a consequence of this result, we can seek to minimize the upper bound $D\left(p_{\mathcal{G}}(x) \| p^T(x)\right)$ by minimizing the sum of terms in (3.65) for some chosen sequence of clique extensions. A greedy approach to minimizing this sum is to try to minimize each term separately, and this type of greedy approach is similar in spirit to the approach taken in [38] for internal multiscale models, although in [38] there is no global cost function such as we have here.

The difficulty with this type of greedy approach is that it is computationally infeasible for many practical problems. This is due to the fact that at least one of the maximal cliques C_i in the sequence of clique extensions must have $M \subseteq C_i$, and consequently, trying to minimize the term $D\left(p(x_{C_i}) \| p_{\mathcal{G}_{i-1}(C_i)}(x_{C_i})\right)$ in (3.65) may be as difficult as solving the original problem. Rather than focus on this greedy approach to approximate realization, we focus instead on the iterative algorithm discussed in the next chapter which does not suffer from this computational difficulty.

Chapter 4

Realizing Approximate Multiscale Models Using EM

IN this chapter, we use the theoretical framework established in the previous chapter to develop an iterative approach for solving the approximate multiscale realization problem. Section 4.1 proposes such an iterative procedure and discusses its convergence properties. As we show, if the iterates of this procedure converge to a fixed density, then this density satisfies the necessary conditions for optimality provided in Theorem 3.8. In Section 4.2, we use the inherent conditional independencies of multiscale models to suggest an efficient recursive procedure for implementing this iterative approach to the realization problem.

From a theoretical point-of-view, the iterative approach presented in Section 4.1 is significant due to its convergence properties, but from a practical point-of-view, such an approach is computationally intractable since each iteration is performed within the space of all densities. In Section 4.3, we address this issue by considering a parameterized subspace of densities, and we show that the iterative approach suggested in Section 4.1 is more commonly known as the EM algorithm when such a parametrization is considered.

While the EM algorithm is computationally viable for many problems of interest, it has one important drawback in that it may not find an optimal solution to the approximate realization problem. As we discuss, though, there is at least one important problem scenario for which the EM algorithm does in practice find optimal solutions, and this occurs when we consider the class of Gaussian multiscale models and when we require the target density $p^*(x_M)$ to be Gaussian as well. Section 4.4 uses this fact to develop a computationally efficient algorithm for solving the Gaussian multiscale realization problem.

■ 4.1 An Iterative Procedure for Solving the Multiscale Realization Problem

In this chapter, we focus on developing an iterative approach to finding solutions to the approximate multiscale realization problem. For the moment, we maintain the perspective of the previous chapter by considering the multiscale realization problem to be a search over all probability densities which factor according to the rooted tree of interest. Later in this chapter, we provide a more practical perspective when we consider the search to be over a parameterized family of densities.

■ 4.1.1 Perspective on the Problem

Recall that approximate multiscale realization problem $\widetilde{Q}$ is defined as follows:

Approximate Multiscale Realization Problem $\widetilde{\mathcal{Q}}$: Find any density $\hat{q} \in \mathcal{P}_{\mathcal{G}_{\preceq}}(V, d)$ which minimizes the cost $D\left(p^*(x_M) \| \hat{q}(x_M)\right)$, *i.e.*

$$\hat{q} = \arg \min_{q \in \mathcal{P}_{\mathcal{G}_{\preceq}}(V,d)} D\left(p^*(x_M) \| q(x_M)\right).$$

In other words, problem $\widetilde{\mathcal{Q}}$ finds a multiscale density $q(x)$ which minimizes the Kullback-Leibler divergence between the target density $p^*(x_M)$ and the marginal density $q(x_M)$. This problem is non-trivial largely because of the fact that the cost function is defined on the subspace $\mathcal{X}_M$, whereas the density of interest, $q(x)$, is defined on the full space $\mathcal{X}$.

In the preceding chapter, we dealt with this difficulty by considering a series of alternative problem formulations $\widetilde{\mathcal{P}}_{\mathcal{G}}^{M}$ each of which involved densities $p(x)$ that satisfied various conditional independence constraints. Consider again one of these problems, which we called problem $\widetilde{\mathcal{P}}^{M}$:

Alternative Approximate Problem $\widetilde{\mathcal{P}}^{M}$: Find any density $\hat{p} \in \mathcal{P}^{M}(V, d)$ which minimizes the cost $D\left(p^*(x_M) \| \hat{p}^{T}(x_M)\right)$, *i.e.*

$$\hat{p} = \arg \min_{p \in \mathcal{P}^{M}(V,d)} D\left(p^*(x_M) \| p^{T}(x_M)\right).$$

In other words, problem $\widetilde{\mathcal{P}}^{M}$ finds any density $p(x)$ which has the target marginal $p^*(x_M)$ and minimizes the KL divergence between $p^*(x_M)$ and $p^{T}(x_M)$. By considering densities $p(x) \in \mathcal{P}^{M}(V, d)$ rather than densities $q(x) \in \mathcal{P}_{\mathcal{G}_{\preceq}}(V, d)$, we have introduced additional degrees of freedom into the optimization problem. In particular, while it is not immediately obvious, these densities $p(x)$ allow the optimization problem to be treated in the full space $\mathcal{X}$ rather than the subspace $\mathcal{X}_M$.

To better understand this point, consider again the decomposition of KL divergence derived in Proposition 3.20. If $q(x) \in \mathcal{P}_{\mathcal{G}_{\preceq}}(V, d)$ is a multiscale density and if $p(x) \in \mathcal{P}^{M}(V, d)$ has the target marginal $p^*(x_M)$, we may write[1]

$$D\left(p^*(x_M) \| q(x_M)\right) = D\left(p(x) \| q(x)\right) - D\left(p(x) \| \mathcal{F}^{M}\left(q(x)\right)\right). \tag{4.1}$$

This decomposition shows that the cost of interest in problem $\widetilde{\mathcal{Q}}$ may be written as the difference of two functions defined on the full space $\mathcal{X}$. In addition, the left-hand-side of (4.1) does not depend on the density $p(x)$, and therefore, $p(x)$ provides additional degrees of freedom in the optimization problem.

The iterative procedure suggested in this section relies on the decomposition in (4.1), and it heavily exploits the degrees of freedom provided by this additional density $p(x)$. Since (4.1) separates the cost function into two terms, this approach seeks to minimize $D\left(p^*(x_M) \| q(x_M)\right)$ by successively optimizing the two terms on the right-hand-side of (4.1). As we soon show, the iterations of this procedure have an extremely simple functional form and are closely related to the theoretical arguments presented in the previous chapter.

Just as in the last chapter, the first two sections of this chapter provide an abstract view of both the realization problem and the iterative approach proposed to solve this problem. The benefit of this point-of-view and the novelty of the ensuing discussion is that significant insight can be gained into the nature of the realization problem and how to solve it. Specifically, the iterates of the

[1] Recall from Section 3.10.2 that $\mathcal{F}^{M}(\cdot)$ maps $q(x)$ to the new density $q(x_{V-M}|x_M)p^*(x_M)$.

approach presented here can be viewed as trading off both the sufficient conditions required for solving the exact realization problem and the necessary conditions for solving the approximate realization problem.[2] In addition, this iterative approach can be viewed as attempting to minimize the upper bound discussed in Section 3.10.4, and as Proposition 3.23 points out, minimizing this upper bound does in fact provide a solution to the approximate realization problem.

While this level of abstraction is beneficial for building intuition about the nature of this problem, there are some drawbacks. First of all, we cannot guarantee that the sequence of densities generated by this iterative approach is guaranteed to converge to a fixed density. This difficulty is an artifact of performing the iterations in the space of all densities and is more of a technicality rather than a practical issue. In fact, this issue is overcome for the most part when we consider searching over parameterized spaces of densities.

A second drawback to this approach is that even if the sequence of densities converges, we are not guaranteed to find an optimal solution to the approximate realization problem. We can only guarantee that the necessary conditions for the approximate realization problem are met. Even when we consider performing the iterations in a parameterized family of densities, this difficulty is present, and as such, for many parameterized problems, the iterations are only guaranteed to converge to either saddle points or local minima of the cost function. This issue can be overcome to some degree by considering different initial starting points.

■ 4.1.2 Alternating Minimizations

Consider the following alternating minimization procedure for finding an optimal solution to the approximate multiscale realization problem. The procedure is initialized with a density $p^{(0)}(x)$ and corresponding multiscale density $q^{(0)} \triangleq \left[p^{(0)}\right]^T$. Then, the following two minimization steps are performed for $i = 1, 2, \ldots$:

$$p^{(i)} \triangleq \arg \min_{p \in \mathcal{P}^M(V,d)} D\left(p(x) \| q^{(i-1)}(x)\right), \tag{4.2a}$$

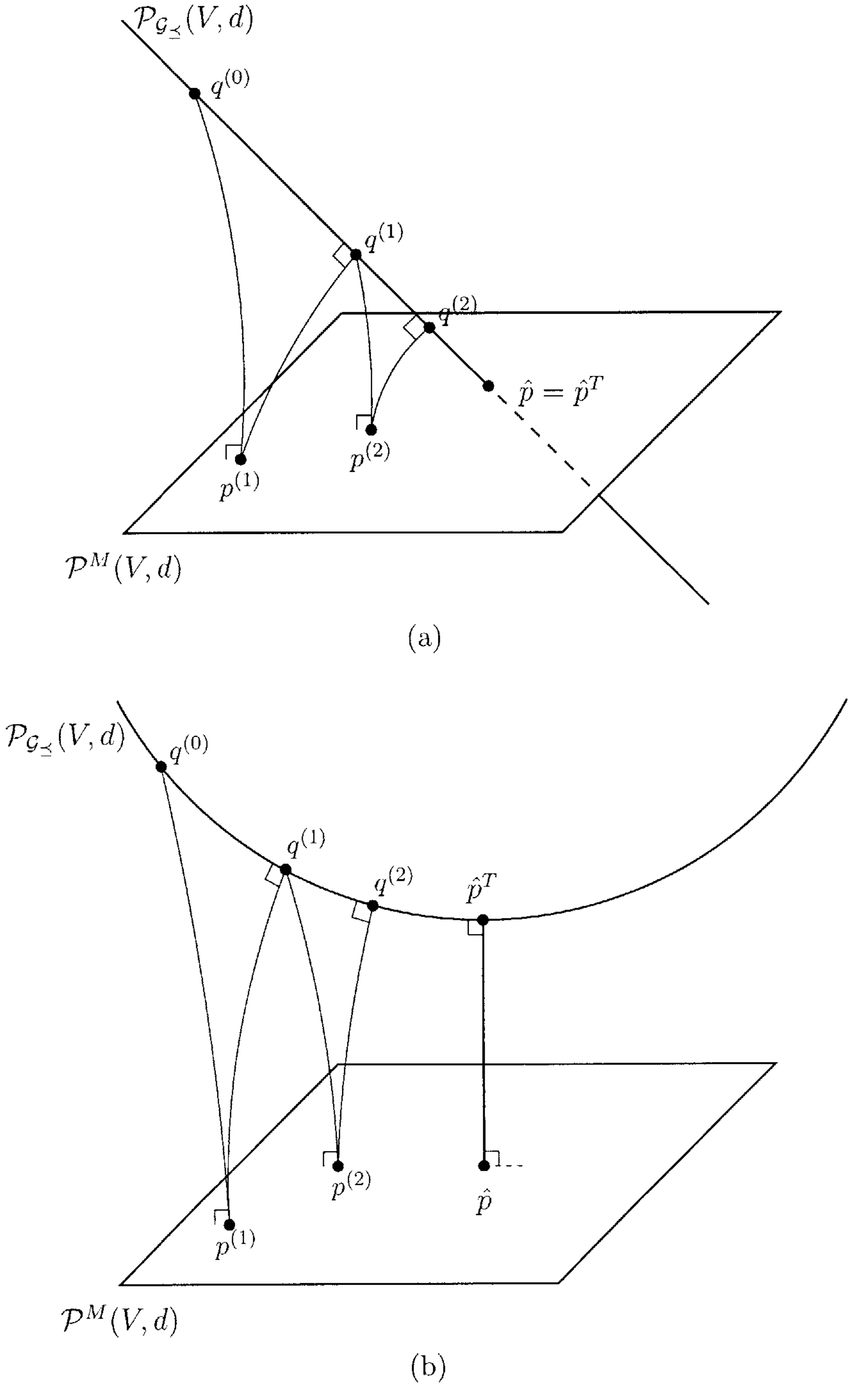
$$q^{(i)} \triangleq \arg \min_{q \in \mathcal{P}_{\mathcal{G}_{\preceq}}(V,d)} D\left(p^{(i)}(x) \| q(x)\right). \tag{4.2b}$$

The first minimization in (4.2a) finds a density $p^{(i)}$ with the correct marginal $p^*(x_M)$, *i.e.* $p^{(i)} \in \mathcal{P}^M(V,d)$, which is "closest" to the multiscale density $q^{(i-1)}$. The second minimization in (4.2b) finds a multiscale density $q^{(i)}$, *i.e.* $q^{(i)} \in \mathcal{P}_{\mathcal{G}_{\preceq}}(V,d)$, which is "closest" to the density $p^{(i)}$ with the correct marginal.

Intuitively, the goal of alternating between the densities $p^{(i)}$ and $q^{(i)}$ in (4.2) is to find a multiscale density $\hat{p}$ with the correct marginal $p^*(x_M)$, *i.e.* $\hat{p}$ simultaneously satisfies the two constraints $\hat{p} \in \mathcal{P}^M(V,d)$ and $\hat{p} \in \mathcal{P}_{\mathcal{G}_{\preceq}}(V,d)$ (equivalently $\hat{p} = \hat{p}^T$). A geometric picture of these two minimization steps is shown in Figure 4.1(a), for the case where an exact solution to the realization problem exists. As previously discussed in Section 3.4.1, the two conditions $\hat{p} \in \mathcal{P}^M(V,d)$ and $\hat{p} = \hat{p}^T$ are sufficient conditions for the exact realization problem. Therefore, when an exact solution exists, the procedure (4.2) seeks to find a point in the intersection of the two sets $\mathcal{P}^M(V,d)$ and $\mathcal{P}_{\mathcal{G}_{\preceq}}(V,d)$.

When an exact solution does not exist, there is no density contained in the intersection of the two sets $\mathcal{P}^M(V,d)$ and $\mathcal{P}_{\mathcal{G}_{\preceq}}(V,d)$, and the procedure (4.2) seeks to mediate the tradeoff between

[2] See Chapter 3 and in particular Section 3.10.4 for a review of these ideas.



(a)

(b)

Figure 4.1. (a) Graphical depiction of alternating minimization procedure (4.2) when an exact solution to the multiscale realization problem exists. (b) Graphical depiction of alternating minimization procedure (4.2) for finding an approximate solution to the multiscale realization problem.

finding a multiscale density and finding a density with the correct marginal $p^*(x_M)$. Figure 4.1(b) provides a geometric picture of the sequence of minimizations in this case.

Based on the discussion in the preceding chapter, we can say much more about the two minimization problems in (4.2). In particular, Corollary 3.5 provides a closed-form expression for a solution to the first minimization problem, and Corollary 3.4 provides a closed-form expression for a solution to the second minimization problem. Using these solutions, the procedure in (4.2) may be written as follows,

$$p^{(i)} = \mathcal{F}^M \left(q^{(i-1)}\right), \tag{4.3a}$$

$$q^{(i)} = \left[p^{(i)}\right]^T. \tag{4.3b}$$

In (4.3a), the multiscale density $q^{(i-1)}$ is transformed into the density $p^{(i)} \in \mathcal{P}^M(V,d)$ via the projective mapping $\mathcal{F}^M(\cdot)$, *i.e.* $p^{(i)}(x) = q^{(i-1)}(x_{V-M}|x_M)p^*(x_M)$. Hence, the first minimization problem is solved by maintaining the same conditional density $q^{(i-1)}(x_{V-M}|x_M)$ and by replacing $q^{(i-1)}(x_M)$ with the desired marginal $p^*(x_M)$. In (4.3b), the density $q^{(i)}$ is obtained by projecting $p^{(i)}$ onto the set of multiscale densities, and as discussed in the previous chapter, this projection is given by $q^{(i)} = \left[p^{(i)}\right]^T$.

In Section 4.1.4, we discuss the convergence properties of the iterations in (4.3), but for the moment, consider what happens if the sequence $p^{(i)}$ does converge to a fixed density $\hat{p}$. Combining (4.3a) and (4.3b) shows that all fixed points $\hat{p}$ satisfy the following,

$$\hat{p} = \mathcal{F}^M \left(\hat{p}^T\right). \tag{4.4}$$

As discussed in Section 3.10.2, the density $\mathcal{F}^M \left(\hat{p}^T\right)$ factors according to all graphs $\mathcal{G}$ which are supergraphs of the tree $\mathcal{G}_{\precsim}$ of interest and in addition have a clique equal to M. As such, the density $\hat{p}$ in (4.4) satisfies $\hat{p} = \hat{p}_{\mathcal{G}}$ for the same graphs $\mathcal{G}$ considered in Lemma 3.9. This lemma indicates that at least one solution to the approximate realization problem must satisfy $\hat{p}_{\mathcal{G}} = \mathcal{F}^M \left(\hat{p}^T\right)$, and consequently, the set of all fixed points of the iterations in (4.3) contains at least one solution to problem $\widetilde{\mathcal{P}}^M$.

By projecting the densities on the left-hand- and right-hand-sides of (4.4) onto the tree $\mathcal{G}_{\preceq}$, a fixed point $\hat{p}$ also satisfies the following

$$\hat{p}^T = \left[\mathcal{F}^M \left(\hat{p}^T\right)\right]^T, \tag{4.5}$$

and the relationship in (4.5) is precisely the necessary condition stated previously in Theorem 3.8. This proves that the fixed points of (4.3) satisfy the necessary conditions for solutions to the approximate realization problem. Furthermore, as evidenced by Proposition 3.23, if any of these fixed points is indeed a solution, then it minimizes the upper bound $D\left(p \| p^T\right)$ over all $p \in \mathcal{P}^M(V,d)$.

■ 4.1.3 Bound Optimization

The alternating minimization procedure in (4.2) or equivalently (4.3) seeks to solve problem $\widetilde{\mathcal{Q}}$ by introducing an extraneous sequence of densities $p^{(i)}$, each of which has the correct marginal $p^*(x_M)$ but none of which are multiscale densities (unless an exact solution is found). As we now discuss, the densities $p^{(i)}$ exploit the additional degrees of freedom present in the decomposition (4.1) discussed earlier. As a result, this particular iterative technique is in fact attempting to minimize the upper bound $D\left(p \| p^T\right)$ - a bound which is tight for solutions to the approximate realization problem.

Consider again the decomposition provided in (4.1). Suppose $q(x)$ is a fixed multiscale density, and consider the second term on the right-hand-side of (4.1). This term can be set to zero by choosing $p(x) = \mathcal{F}^M(q(x))$, and as a result, $D\left(p^*(x_M) \| q(x_M)\right) = D\left(p(x) \| q(x)\right)$ for this choice of $p(x)$. Notice that the first step of the iterative procedure in (4.3a) makes a similar choice for the density $p^{(i)}(x)$. Consider now the problem of minimizing the first term on the right-hand-side of (4.1) with respect to all multiscale densities $q(x)$. Corollary 3.4 indicates that this is accomplished by choosing $q(x) = p^T(x)$. Notice that this choice for $q(x)$ is identical to the choice for $q^{(i)}(x)$ in the second step of the iterative procedure in (4.3b).

Based on these observations, we argue that the iterative procedure (4.3) successively minimizes the two terms on the right-hand-side of (4.1); the first step chooses $p(x)$ to minimize the second term, while the second step chooses $q(x)$ to minimize the first term. Or, said in a different way, the first step seeks to satisfy the necessary conditions for solutions to the approximate realization problem as stated in Theorem 3.8, while the second step seeks to satisfy the sufficient conditions for solutions to the exact realization problem as stated in Theorem 3.3.

At first glance, it may seem counter-intuitive to minimize the second term in (4.1) because after all, increasing its value helps to decrease the overall cost function. However, the importance of setting the second term to zero is that at each iteration we are decreasing an upper bound on the cost function. To validate this claim, consider the sequence of densities $p^{(i)}$ and $q^{(i)}$, as well as the two minimization problems in (4.2). From the first minimization problem, we know that $D\left(p^{(i)} \| q^{(i-1)}\right) \leq D\left(p \| q^{(i-1)}\right)$ for all $p \in \mathcal{P}^M(V,d)$, including $p = p^{(i-1)}$. From the second minimization problem, we know that $D\left(p^{(i)} \| q^{(i)}\right) \leq D\left(p^{(i)} \| q\right)$ for all $q \in \mathcal{P}_{\mathcal{G}_{\preceq}}(V,d)$, including $q = q^{(i-1)}$. Using these two facts, we can then write the following sequence of inequalities,

$$D\left(p^{(i)} \| q^{(i)}\right) \leq D\left(p^{(i)} \| q^{(i-1)}\right) \leq D\left(p^{(i-1)} \| q^{(i-1)}\right).$$

Finally, using only the first and last terms from above, as well as the definition of $q^{(i)}$ and $q^{(i-1)}$ in (4.3b), gives the following important inequality

$$D\left(p^{(i)} \| \left[p^{(i)}\right]^T\right) \leq D\left(p^{(i-1)} \| \left[p^{(i-1)}\right]^T\right). \tag{4.6}$$

This inequality proves that the sequence of densities $p^{(i)}$ decreases (or at least does not increase) the cost $D\left(p \| p^T\right)$ at each iteration.

Based on the preceding inequalities, we see that the iterations in (4.2) seek a minimum of the function $D\left(p \| p^T\right)$ over all $p \in \mathcal{P}^M(V,d)$. Furthermore, $D\left(p \| p^T\right)$ is an upper bound for the function $D\left(p^*(x_M) \| p^T(x_M)\right)$, a fact which can be seen by substituting $q(x) = p^T(x)$ into (4.1). Recall that this upper bound was considered in the previous chapter. In particular, Proposition 3.23 proved that minimizing this bound over the space of densities $p \in \mathcal{P}^M(V,d)$ is guaranteed to generate a solution to the approximate realization problem. However, Proposition 3.23 does not suggest than any density $\hat{p}$ satisfying $\hat{p} = \mathcal{F}^M\left(\hat{p}^T\right)$ is necessarily a minimizer of this bound, and therefore, as previously mentioned, we cannot guarantee that all fixed points are indeed solutions to the approximate realization problem.

■ 4.1.4 Convergence Properties

As a final part of our discussion of (4.2) and (4.3), we analyze the convergence properties of this type of iterative scheme. In the previous section, we showed that the sequence $p^{(i)}$ decreases an upper bound at each iteration, as evidenced by the inequality in (4.6). In this section, we examine the sequence $q^{(i)}$ with respect to the cost function $D\left(p^*(x_M)\|q^{(i)}(x_M)\right)$, *i.e.* the cost to be minimized in the approximate realization problem. As we show, this cost is decreased with each iteration.

To aid in this analysis, define the sequence of non-negative real numbers ε_i, $i = 1, 2, \ldots$, as follows

$$\varepsilon_i \triangleq D\left(p^*(x_M)\|q^{(i)}(x_M)\right). \tag{4.7}$$

Our goal is to show that the sequence $\{\varepsilon_i\}$ is non-increasing. Using (4.1) with the choice $p(x) = p^{(i+1)}(x)$, we obtain an equivalent definition of ε_i,

$$\varepsilon_i = D\left(p^{(i+1)}\|q^{(i)}\right) - D\left(p^{(i+1)}\|\mathcal{F}^M\left(q^{(i)}\right)\right) = D\left(p^{(i+1)}\|q^{(i)}\right), \tag{4.8}$$

where we have used the fact that $p^{(i+1)} = \mathcal{F}^M\left(q^{(i)}\right)$. Using Proposition 3.19, (4.8) may be decomposed as follows,

$$\begin{aligned}\varepsilon_i &= D\left(p^{(i+1)}\|\left[p^{(i+1)}\right]^T\right) + D\left(\left[p^{(i+1)}\right]^T\|q^{(i)}\right)\\ &= D\left(p^{(i+1)}\|q^{(i+1)}\right) + D\left(q^{(i+1)}\|q^{(i)}\right).\end{aligned} \tag{4.9}$$

Calling upon (4.1) to further decompose $D\left(p^{(i+1)}\|q^{(i+1)}\right)$ in (4.9), provides the final decomposition of interest,

$$\begin{aligned}\varepsilon_i &= D\left(p^*(x_M)\|q^{(i+1)}(x_M)\right) + D\left(p^{(i+1)}\|\mathcal{F}^M\left(q^{(i+1)}\right)\right) + D\left(q^{(i+1)}\|q^{(i)}\right)\\ &= \varepsilon_{i+1} + D\left(p^{(i+1)}\|p^{(i+2)}\right) + D\left(q^{(i+1)}\|q^{(i)}\right).\end{aligned} \tag{4.10}$$

Due to the non-negativity property of the KL divergence, (4.10) indicates that $\varepsilon_{i+1} \leq \varepsilon_i$. Therefore, the sequence of real numbers $\{\varepsilon_i\}$ is non-increasing for $i = 1, 2, \ldots$, and furthermore, the sequence is bounded below by 0. These two facts imply that the sequence $\{\varepsilon_i\}$ converges to some real number $\bar{\varepsilon} \geq 0$ [91]. Consequently, $\varepsilon_i - \varepsilon_{i+1} \longrightarrow 0$ as $i \longrightarrow \infty$, which implies that both of the terms $D\left(p^{(i+1)}\|p^{(i+2)}\right)$ and $D\left(q^{(i+1)}\|q^{(i)}\right)$ in (4.10) approach zero as $i \longrightarrow \infty$. These observations then indicate that there exist densities $\hat{p}$ and $\hat{q}$ such that $p^{(i)} \longrightarrow \hat{p}$ and $q^{(i)} \longrightarrow \hat{q}$ almost everywhere.[3]

In summary, this section suggests an interesting iterative approach to solving the approximate realization problem. We have discussed this iterative scheme in three different contexts:

[3] This statement must be qualified due to several technicalities. First of all, the decomposition in (4.10) was derived assuming that all of the KL divergences are finite. It is possible that some density $p^{(i)}$ or $q^{(i)}$ may not satisfy the measure-theoretic conditions for the KL divergence to be finite. Even if this is not the case, it is possible that the sequence $p^{(i)}$ converges to a density where in the limit the decomposition in (4.10) does not hold. Second, if (4.10) does hold for all $i = 1, 2, \ldots$, then the sequence $p^{(i)}$ converges to a fixed density $\hat{p}$ almost everywhere. This means that some portion of the density $\hat{p}$ may not be invariant under the mapping $\mathcal{F}^M\left(\hat{p}^T\right)$ but only on a set of measure zero.

Both of these issues are not a concern for most practical problems. By placing additional restrictions on the class of densities under consideration, *e.g.* they satisfy certain smoothness conditions, these technicalities are no longer a concern. However, we do not investigate these issues further since they are beyond the scope of this thesis.

(1) alternating between the minimization problems in (4.2);

(2) minimizing the upper bound $D\left(p \| p^T\right)$;

(3) minimizing the cost function $D\left(p^*(x_M) \| q(x_M)\right)$.

The first context shows that the iterations simultaneously tradeoff two constraints: (i) the density has the marginal $p^*(x_M)$ and (ii) the density is a multiscale density. The second context helps to relate the ideas discussed here to the results presented in the previous chapter. The final context shows that the iterations in (4.3) seek a minimum of the cost function of interest and that under some restrictions these iterations converge in the limit to a density satisfying the necessary conditions for optimality.

4.2 Taking Advantage of Conditional Independence Structure

In this section, we continue our discussion of the iterative procedure suggested in the previous section. By taking advantage of the conditional independencies exhibited by multiscale models, we show how to reduce the computational complexity of the calculations required to implement the iterations in (4.3). In particular, we suggest a recursive procedure for calculating the marginal densities needed to perform each iteration. As we show, the proposed recursion can be divided into two separate steps: (1) incorporating the target density and (2) calculating conditional densities. The first step, discussed in Section 4.2.1, involves the merging of the target density $p^*(x_M)$ with the conditional density $q^{(i-1)}(x_{V-M}|x_M)$ to give $p^{(i)}(x)$ in (4.3a). The second step, discussed in Section 4.2.2, involves the calculation of the conditional density $q^{(i-1)}(x_{V-M}|x_M)$ necessary to perform the first step.

The ideas presented here are similar to those widely discussed in the context of Kalman filtering [60] and smoothing [86], belief propagation [84], and tree reparametrization [105, 106], which we point out in the ensuing discussion. However, there is an important distinction between each of these contexts and the ideas presented here, due to the fact that a target density $p^*(x_M)$ has been specified. Because of this, we must include an additional step, namely incorporating the target density $p^*(x_M)$, in our recursive procedure.

4.2.1 Incorporating the Target Density

From an abstract point-of-view, the procedure in (4.3) requires the two densities $p^{(i)}$ and $q^{(i)}$ to be calculated at each iteration. As we now demonstrate, it is not necessary to calculate these densities in their entirety in order to implement this procedure; rather, it is sufficient to calculate specific marginals of $p^{(i)}$ and $q^{(i)}$. To gain some insight, consider first the projection $q^{(i)} = \left[p^{(i)}\right]^T$ in (4.3b), which forms the multiscale density $q^{(i)}$ with respect to a specified rooted tree $\mathcal{G}_{\preceq}$. Recall from Section 3.3.3 that this projection can be written as follows,

$$q^{(i)}(x) = p^{(i)}(x_{v_0}) \prod_{v \in V - \{v_0\}} \frac{p^{(i)}(x_v, x_{\pi(v)})}{p^{(i)}(x_{\pi(v)})}, \tag{4.11}$$

where v_0 is the root vertex of $\mathcal{G}_{\preceq}$. As (4.11) shows, the density $q^{(i)}$ can be calculated if all of the marginals $p^{(i)}(x_v, x_{\pi(v)})$, $v \in V - \{v_0\}$, are known. Equivalently, if we consider the undirected

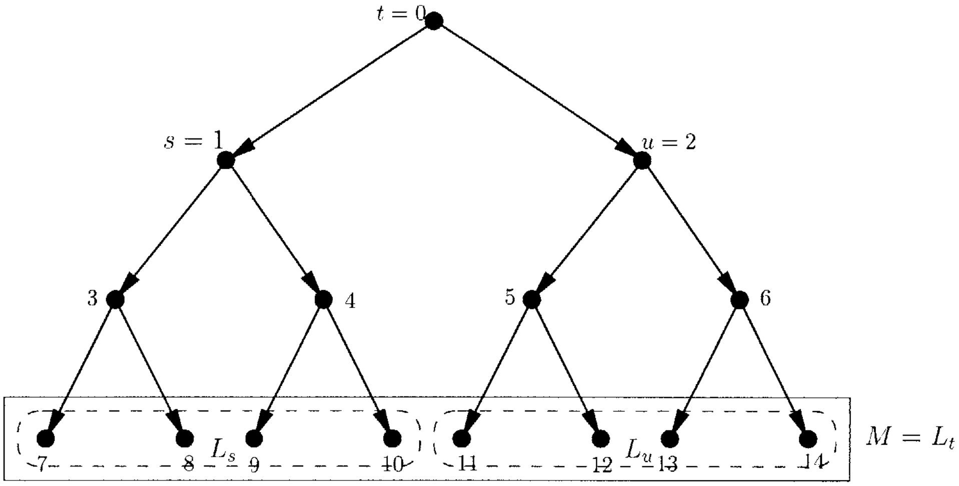


Figure 4.2. A rooted tree $\mathcal{G}_{\preceq}$ where vertex t is the parent of vertex s and the marginalization constraint set M is precisely the set of leaf vertices of $\mathcal{G}_{\preceq}$, as indicated by the solid box. The leaf vertices which descend from vertices s and u are denoted respectively by the sets L_s and L_u, as indicated by the dashed boxes.

tree $\mathcal{G}_{\preceq}^{\sim} = (V, E)$, calculating $q^{(i)}$ requires knowledge of the marginals $p^{(i)}(x_s, x_t)$ along each edge $(s,t) \in E$. Therefore, the multiscale density $q^{(i)}$ may be determined as long as a relatively small subset of the marginals of $p^{(i)}$ are known. We now demonstrate that these marginals may be calculated in a recursive fashion.

For notational simplicity, we drop the superscript i; equivalently, the reader can assume $q(x) \triangleq q^{(i-1)}(x)$ and $p(x) \triangleq p^{(i)}(x)$. Also, to simplify our discussion we henceforth assume that the marginalization constraint set $M \subset V$ is precisely equal to the leaf vertices of the rooted tree $\mathcal{G}_{\preceq}$ of interest.[4] Consider two vertices s and t such that $(s,t) \in E$ and $s, t \notin M$, and without loss of generality, assume that t is the parent of s in the rooted tree $\mathcal{G}_{\preceq}$. Figure 4.2 provides an example of such a choice for s and t. Now, suppose a multiscale density $q(x)$ and a target density $p^*(x_M)$ are given, and consider the marginal $p(x_s, x_t)$ of $p(x) \triangleq q(x_{V-M}|x_M)p^*(x_M)$. This marginal may be calculated as follows,

$$\begin{aligned} p(x_s, x_t) &= \int_{\mathcal{X}_{V-\{s,t\}}} p(x)dx = \int_{\mathcal{X}_{V-\{s,t\}}} q(x_{V-M}|x_M)p^*(x_M)dx \\ &= \int q(x_s, x_t|x_M)p^*(x_M)dx_M, \end{aligned} \tag{4.12}$$

where the last equation follows from the fact that $s, t \notin M$.

Recall that the set L_s contains all leaf vertices of a rooted tree $\mathcal{G}_{\preceq}$ which descend from vertex s. For example, Figure 4.2 shows that the leaf vertices $L_s = \{7, 8, 9, 10\}$ are descendants of vertex $s = 1$. This figure also demonstrates the fact that vertex t graphically separates the collection of

[4] The recursive procedure described in this section and the next may be extended to include the problem where M contains non-leaf vertices or extended to include the state augmentation problem where an augmented marginalization set $M^\sharp$ is specified (see Section 2.7.3 for details of the state augmentation problem). Neither of these generalizations is discussed here.

vertices $\{s\} \cup L_s$ from the remaining leaf vertices not contained in L_s. Consequently, using the fact that $q(x)$ is a multiscale density, we may write $q(x_s|x_t, x_M) = q(x_s|x_t, x_{L_s})$, which may then be used to simplify the expression in (4.12),

$$p(x_s, x_t) = \int q(x_s|x_t, x_M) q(x_t|x_M) p^*(x_M) dx_M = \int q(x_s|x_t, x_{L_s}) q(x_t|x_M) p^*(x_M) dx_M.$$

Notice that $q(x_t|x_M)p^*(x_M)$ in the final expression is by definition equal to $p(x_t, x_M)$. Using this fact and integrating out the variables x_{M-L_s} provides an important expression for the marginal $p(x_s, x_t)$,

$$p(x_s, x_t) = \int q(x_s|x_t, x_{L_s}) p(x_t, x_M) dx_M = \int q(x_s|x_t, x_{L_s}) p(x_t, x_{L_s}) dx_{L_s}. \tag{4.13}$$

The final equality in (4.13) demonstrates that $p(x_s, x_t)$ may be calculated if the conditional density $q(x_s|x_t, x_{L_s})$ and the marginal $p(x_t, x_{L_s})$ are known. This suggests a recursive procedure for calculating the necessary marginals $p(x_s, x_t)$. The recursion is initialized by setting t equal to the root vertex and computing $p(x_t, x_M) = q(x_t|x_M)p^*(x_M)$. For any child vertex s of t, the marginal $p(x_t, x_{L_s})$ may be calculated by integrating $p(x_t, x_M)$. Then, the density $p(x_s, x_t, x_{L_s})$ may be calculated as follows, where the same sequence of steps is used as was used to derive (4.13),

$$\begin{aligned} p(x_s, x_t, x_{L_s}) &= \int q(x_s, x_t|x_M) p^*(x_M) dx_{M-L_s} = \int q(x_s|x_t, x_M) q(x_t|x_M) p^*(x_M) dx_{M-L_s} \\ &= \int q(x_s|x_t, x_{L_s}) p(x_t, x_M) dx_{M-L_s} \\ &= q(x_s|x_t, x_{L_s}) p(x_t, x_{L_s}). \end{aligned} \tag{4.14}$$

Assuming $q(x_s|x_t, x_{L_s})$ is known, (4.14) shows how to calculate $p(x_s, x_t, x_{L_s})$ from $p(x_t, x_{L_s})$. By integrating out the variables x_{L_s} from $p(x_s, x_t, x_{L_s})$, the desired density $p(x_s, x_t)$ can then be obtained. Notice that performing this integration on the right-hand-side of (4.14) gives the same expression originally derived in (4.13).

The purpose of calculating the density $p(x_s, x_t, x_{L_s})$ in (4.14), besides providing the marginal $p(x_s, x_t)$, is that x_t may be marginalized away to give $p(x_s, x_{L_s})$. Then, using the marginal $p(x_s, x_{L_s})$, all edge marginals $p(x_u, x_v)$ within the subtree descending from s may be calculated in the same recursive manner. Therefore, the density $p(x_s, x_{L_s})$ provides sufficient information to the subtree descending from vertex s, just as the density $p(x_t, x_{L_t})$ provides sufficient information to the subtree descending from vertex t. The following example provides a specific illustration of this recursive procedure.

Example 4.1 (Recursive Procedure for Incorporating the Target Density).
Consider the rooted tree $\mathcal{G}_{\preceq}$ shown in Figure 4.2, and assume a target density $p^*(x_7, \ldots, x_{14}) = p^*(x_M)$ defined on the leaf vertices of $\mathcal{G}_{\preceq}$ is given. This example shows how the result in (4.14) may be used to recursively calculate the set of marginal densities $p(x_s, x_t)$ along each edge of $\mathcal{G}_{\preceq}$.

The recursion is initialized by computing the density $p(x_0, x_M) = q(x_0|x_M)p^*(x_M)$. Based on this initial density, we have sufficient information to calculate the two marginals $p(x_0, x_1)$ and $p(x_0, x_2)$. To calculate $p(x_0, x_1)$, we first marginalize the density $p(x_0, x_M) = p(x_0, x_7, \ldots, x_{14})$ to

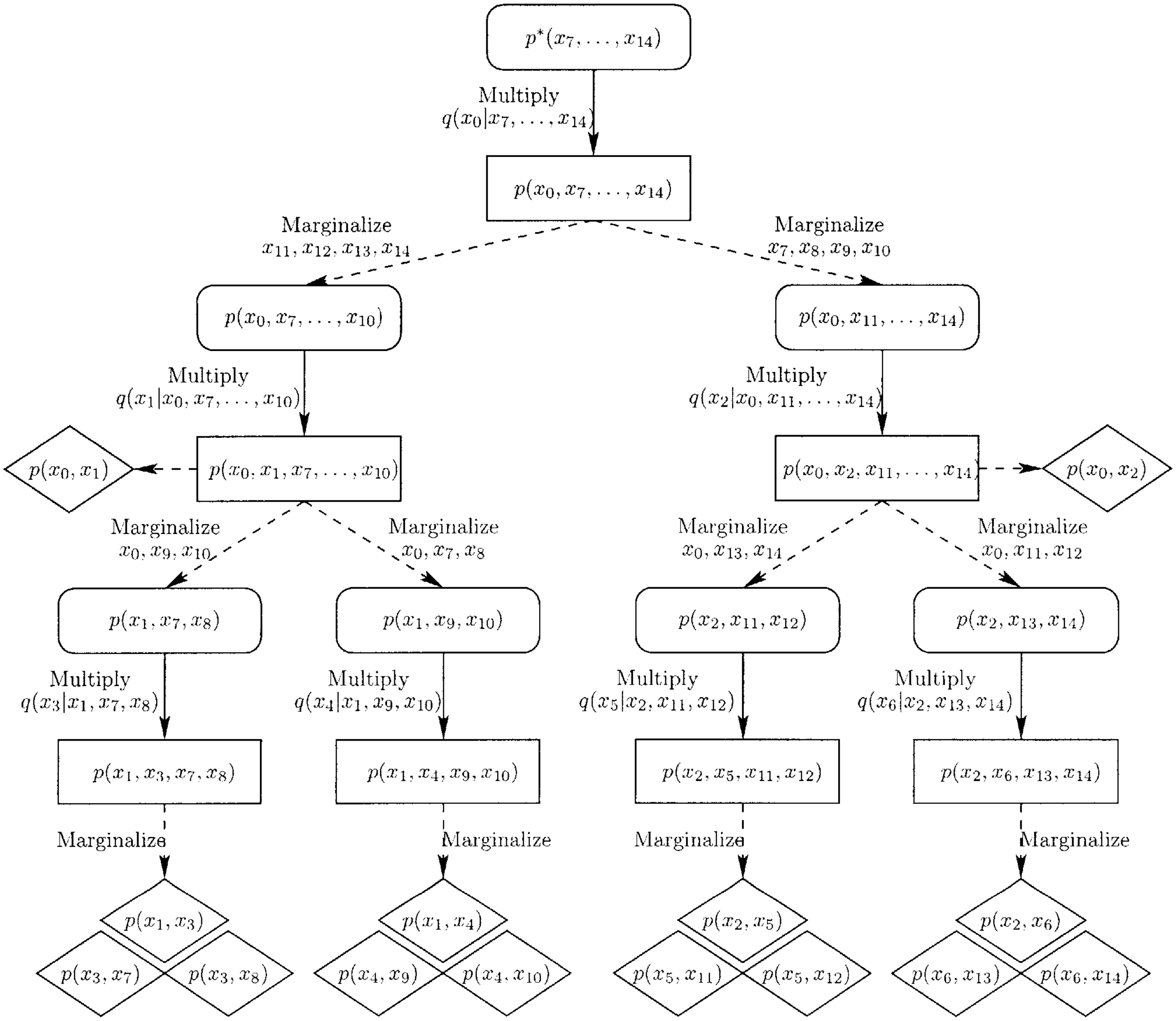


Figure 4.3. Block diagram illustrating a recursive approach to calculating marginals $p(x_s, x_t)$ along all edges in the graph $\mathcal{G}_{\preceq}$ shown in Figure 4.2. The rectangular boxes contain the marginal densities generated by the recursion in (4.14), while the rounded boxes contain intermediate densities which result from a marginalization step. The diamond-shaped boxes contain the marginals $p(x_s, x_t)$ of interest.

give $p(x_0, x_{L_1}) = p(x_0, x_7, x_8, x_9, x_{10})$. Then, the result in (4.14) indicates that $p(x_0, x_1, x_{L_1}) = p(x_0, x_1, x_7, x_8, x_9, x_{10})$ can be determined as follows,

$$p(x_0, x_1, x_7, x_8, x_9, x_{10}) = q(x_1|x_0, x_7, x_8, x_9, x_{10})p(x_0, x_7, x_8, x_9, x_{10}) = q(x_1|x_0, x_{L_1})p(x_0, x_{L_1}).$$

Finally, the density $p(x_0, x_1, x_7, x_8, x_9, x_{10})$ can be integrated to give the marginal $p(x_0, x_1)$ of interest, and the marginals $p(x_1, x_7, x_8)$ and $p(x_1, x_9, x_{10})$ can also be computed for use in subsequent calculations.

This process continues in a similar fashion until all of the relevant marginals $p(x_s, x_t)$ are computed. The block diagram in Figure 4.3 graphically illustrates the calculations required for this example. The densities in the square boxes represent the marginals calculated using (4.14), while the densities in the rounded boxes represent intermediate marginalization steps. The diamond-shaped boxes contain the edge marginals $p(x_s, x_t)$ of interest.

One interesting aspect of this block diagram is that it illustrates the similarity between the iterative approach suggested here for the approximate realization problem and the sequential approach suggested in Section 2.7.2 for the exact realization problem. Specifically, Example 2.7 considers the exact realization problem for the same graph considered in this example, and Figure 2.12 demonstrates that the structure of the two algorithms is identical. The only difference is that the so-called design density $\bar{p}(\cdot)$ has been replaced by the multiscale density $q(\cdot)$. Therefore, within the context of the iterative approach discussed here, the multiscale density $q(\cdot)$ provides a convenient guess for the design density $\bar{p}(\cdot)$. ◀

As the preceding example illustrates, the process of introducing the target density $p^*(x_M)$ begins by calculating the density $p(x_{v_0}, x_M) = q(x_{v_0}|x_M)p^*(x_M)$, where v_0 is the root of the tree. Subsequent calculations are then performed to generate densities which include the child vertices of v_0, and so forth. In fact, the recursive calculation in (4.14) is well-defined as long as the marginal $p(x_t, x_{L_t})$ has been calculated before any child of vertex t is considered. In other words, the recursion is well-defined as long as we consider a top-down ordering $(v_1, \ldots, v_m)$ on the non-leaf vertices of $\mathcal{G}_{\preceq}$.[5] Given this fact, we now formally state the recursive algorithm proposed in this section.

Algorithm 4.1 (Incorporating the Target Density).
Let $\mathcal{G}_{\preceq}$ be a given rooted tree. Assume M is precisely the set of leaf vertices of $\mathcal{G}_{\preceq}$ and that a target density $p^*(x_M)$ is specified. Also, assume that a multiscale density $q(x)$ is given and that all marginals of $q(x)$ can be calculated. Choose any top-down ordering $(v_1, \ldots, v_m)$ on the non-leaf vertices of $\mathcal{G}_{\preceq}$.

```
Set $t = v_1$, and compute $p(x_t, x_{L_t}) = p(x_t, x_M) = q(x_t|x_M)p^*(x_M)$.
FOR $i = 2, \ldots, m$ DO:
    (1) Set $s = v_i$, and set $t$ equal to the unique parent of $s$.
    (2) Marginalize $p(x_t, x_{L_t})$ to get $p(x_t, x_{L_s})$.
    (3) Compute $p(x_s, x_t, x_{L_s}) = q(x_s|x_t, x_{L_s})p(x_t, x_{L_s})$.
    (4) Marginalize $p(x_s, x_t, x_{L_s})$ to get $p(x_s, x_t)$ and $p(x_s, x_{L_s})$.   ◀
```

[5] Recall from Definition 2.9 that a top-down ordering $(v_1, \ldots, v_m)$ places the root vertex first. Furthermore, the parent of vertex v_i must appear before v_i in the ordering.

The fact that Algorithm 4.1 is defined for all top-down orderings of the non-leaf vertices raises the important question of considering other vertex orderings. The answer to this question is that a similar algorithm exists for any such ordering; however, the recursion derived in (4.14) differs in form depending on the chosen ordering. For convenience, we focus on Algorithm 4.1 since the derived recursion is particularly simple.

4.2.2 Computational Structure for Tree-Based Conditional Densities

The recursive procedure suggested in the previous section provides a means to implement the first step in (4.3) without calculating the entire density $p(x)$. In addition, the third step of Algorithm 4.1 shows that it is not necessary to calculate the entire conditional density $q(x_{V-M}|x_M)$. Instead, only certain pieces of this conditional density are required, namely $q(x_{v_0}|x_M)$ for the root vertex and $q(x_s|x_t, x_{L_s})$ for every non-leaf and non-root vertex s and its parent t. So far, we have assumed that all of these conditional densities are readily available. However, within the context of the iterative procedure (4.3), the multiscale density $q^{(i)}(x)$ changes with each iteration, and therefore, it is important that these conditional densities be calculated efficiently. This section provides a recursive procedure for calculating these densities – a procedure which takes advantage of the conditional independencies exhibited by multiscale models.

Consider the conditional density $q(x_s|x_t, x_{L_s})$ required in the third step of Algorithm 4.1. Using the chain rule for probabilities, this density may be rewritten as follows,

$$q(x_s|x_t, x_{L_s}) = \frac{q(x_s, x_t|x_{L_s})}{q(x_t|x_{L_s})} = \frac{q(x_t|x_s, x_{L_s})q(x_s|x_{L_s})}{q(x_t|x_{L_s})}.$$

Since vertex s separates vertex t from all of the vertices in the set L_s, the conditional density $q(x_t|x_s, x_{L_s})$ is equal to $q(x_t|x_s)$, which then gives the following expression,

$$q(x_s|x_t, x_{L_s}) = \frac{q(x_t|x_s)q(x_s|x_{L_s})}{q(x_t|x_{L_s})}. \tag{4.15}$$

This expression forms the basis of the recursion suggested here because it shows how to combine the terms $q(x_s|x_{L_s})$ and $q(x_t|x_{L_s})$ to form $q(x_s|x_t, x_{L_s})$.[6] The remainder of our discussion focuses on how to calculate $q(x_s|x_{L_s})$ and $q(x_t|x_{L_s})$ efficiently.

First of all, calculating $q(x_t|x_{L_s})$ is straightforward if the density $q(x_s|x_{L_s})$ is known. This follows directly from the chain rule for probabilities and the Markov properties of multiscale models,

$$\begin{aligned} q(x_t|x_{L_s}) &= \int q(x_s, x_t|x_{L_s})dx_s = \int q(x_s|x_{L_s})q(x_t|x_s, x_{L_s})dx_s \\ &= \int q(x_s|x_{L_s})q(x_t|x_s)dx_s. \end{aligned} \tag{4.16}$$

Within the context of the Kalman filter, the expression in (4.16) is often called the *prediction step.* Namely, the best "guess" of vector X_t given an observation $X_{L_s} = x_{L_s}$ is predicted based on the best "guess" for vector X_s and the local conditional density $q(x_t|x_s)$. It is important to note that in our context the density $q(x_t|x_{L_s})$ is a function of both x_t and x_{L_s}, *i.e.* we consider all possible outcomes $x_{L_s} \in \mathcal{X}_{L_s}$. This is necessary in order to perform the integration with the density $p^*(x_M)$.

[6] The density $q(x_t|x_s)$ is readily available because it is a marginal along an edge of the multiscale model.

When given a single observation as is the case in applications involving observed data, the value of $X_{L_s} = x_{L_s}$ is fixed and $q(x_t|x_{L_s})$ is simply a function of x_t.

Recall that $\chi(t)$ is the set containing all child vertices of t in the tree $\mathcal{G}_{\preceq}$. We now show how $q(x_t|x_{L_t})$ can be calculated if $q(x_t|x_{L_s})$ is known for all $s \in \chi(t)$. Using Bayes' rule on $q(x_t|x_{L_t})$ and introducing the variables $x_{\chi(t)}$ gives the following,

$$q(x_t|x_{L_t}) = \frac{q(x_{L_t}|x_t)q(x_t)}{q(x_{L_t})} = \frac{q(x_t)}{q(x_{L_t})} \int q(x_{\chi(t)}, x_{L_t}|x_t) dx_{\chi(t)}. \tag{4.17}$$

The Markov properties of the multiscale model may be used to simplify the preceding expression. As an example of this fact, consider the tree in Figure 4.2, and notice that vertex $t = 0$ graphically separates the two sets of vertices $\{s\} \cup L_s = \{1, 7, 8, 9, 10\}$ and $\{u\} \cup L_u = \{2, 11, 12, 13, 14\}$, where s and u are the only child vertices of t, *i.e.* $\chi(t) = \{s, u\}$. Because of this graphical separation, the Markov properties of the multiscale model indicate that the density $q(x_{\chi(t)}, x_{L_t}|x_t) = q(x_s, x_u, x_{L_s}, x_{L_u}|x_t)$ may be written as the product of the two densities $q(x_s, x_{L_s}|x_t)$ and $q(x_u, x_{L_u}|x_t)$.

More generally, in any tree, vertex t graphically separates the collection of vertices $\{s_1\} \cup L_{s_1}, \ldots,$ $\{s_q\} \cup L_{s_q}$, where $s_i \in \chi(t)$, and consequently, we can write $q(x_{\chi(t)}, x_{L_t}|x_t) = \prod_{s \in \chi(t)} q(x_s, x_{L_s}|x_t)$. Using this factorization in the expression in (4.17) gives the following,

$$q(x_t|x_{L_t}) = \frac{q(x_t)}{q(x_{L_t})} \int \left[\prod_{s \in \chi(t)} q(x_s, x_{L_s}|x_t) \right] dx_{\chi(t)} = \frac{q(x_t)}{q(x_{L_t})} \prod_{s \in \chi(t)} q(x_{L_s}|x_t).$$

Using Bayes' rule on the terms $q(x_{L_s}|x_t)$ provides the final expression of interest,

$$\begin{aligned} q(x_t|x_{L_t}) &= \frac{q(x_t)}{q(x_{L_t})} \prod_{s \in \chi(t)} \frac{q(x_t|x_{L_s})q(x_{L_s})}{q(x_t)} \\ &= \left[\frac{\prod_{s \in \chi(t)} q(x_{L_s})}{q(x_{L_t})} \right] \cdot \left[\frac{1}{q(x_t)^{|\chi(t)|-1}} \right] \cdot \prod_{s \in \chi(t)} q(x_t|x_{L_s}). \end{aligned} \tag{4.18}$$

Within the context of the Kalman filter (as generalized to multiscale models in [13]), the expression in (4.18) is often called the *merge step*. This is due to the fact that the best "guesses" for X_t based on the separate observations $X_{L_s} = x_{L_s}$, $s \in \chi(t)$, are merged to form the best "guess" for X_t based on the entire observation $X_{L_t} = x_{L_t}$. The essence of the merge step is accomplished by the final term in (4.18), which computes the product of the densities $q(x_t|x_{L_s})$. The first two terms in (4.18) provide the necessary correction factors to make $q(x_t|x_{L_t})$ a density. It is important to note that calculating the first term in (4.18) can be problematic for implementing the merge step in the abstract sense discussed here. This is because the densities $q(x_{L_t})$ and $q(x_{L_s})$ involved in this term are not edge marginals and are therefore not easily accessible. However, when conditioning on a single outcome $X_{L_t} = x_{L_t}$ rather than all possible outcomes $x_{L_t} \in \mathcal{X}_{L_t}$, the first term in (4.18) is simply a constant and provides the scaling necessary to make $q(x_t|x_{L_t})$ a density.[7] In the Gaussian case considered later, this term does not pose a problem since the form of the density $q(x_t|x_{L_t})$ can be determined using only the latter two terms in (4.18).

[7] In applications involving data, it is only necessary to condition on a single outcome or a finite set of outcomes, and in this situation, the constant term in (4.18) can be ignored.

The three relationships derived in (4.15), (4.16), and (4.18) form the basis of a recursive procedure for calculating the conditional densities required to implement Algorithm 4.1. The procedure begins with a pass of the tree which starts at the leaf vertices and works toward the root – a pass which is used to recursively calculate the densities $q(x_s|x_{L_s})$ and $q(x_t|x_{L_s})$. The calculations required for this pass are well-defined as long as a bottom-up ordering on the non-leaf vertices is chosen,[8] and the recursive equations in (4.16) and (4.18) are used to implement these calculations. Then, a subsequent pass of the tree, which starts at the root and works toward the leaf vertices, is used to calculate $q(x_s|x_t, x_{L_s})$, and this is accomplished by choosing a top-down ordering on the non-leaf vertices and using the relationship in (4.15). This two-sweep process is summarized below in Algorithm 4.2.

Algorithm 4.2 (Calculating Tree-Based Conditional Densities).
Let $\mathcal{G}_{\preceq}$ be a given rooted tree, and assume a multiscale density $q(x)$ is given.

```
Initialization: For each leaf vertex $s$ and its unique parent $t$, $q(x_t|x_{L_s}) \triangleq q(x_t|x_s)$.

Upward Pass: Choose any bottom-up ordering $(u_1, \dots, u_m)$ on the non-leaf vertices of $\mathcal{G}_{\preceq}$.
FOR $i = 1, \dots, m$ DO:
    (1) Set $t = u_i$.
    (2) Compute $q(x_t|x_{L_t})$ using (4.18).
    (3) Compute $q(x_{\pi(t)}|x_{L_t}) = \int q(x_t|x_{L_t}) q(x_{\pi(t)}|x_t) dx_t$.

Downward Pass: Choose any top-down ordering $(v_1, \dots, v_m)$ on the non-leaf vertices of $\mathcal{G}_{\preceq}$.
FOR $i = 2, \dots, m$ DO:
    (1) Set $s = v_i$, and set $t$ equal to the unique parent of $s$.
    (2) Compute $q(x_s|x_t, x_{L_s})$ using (4.15).                                    ◀
```

Figure 4.4(a) provides a graphical illustration of the upward pass described in Algorithm 4.2. For each child vertex $s_i \in \chi(t)$, the prediction step is used to compute $q(x_t|x_{L_{s_i}})$ based on the density $q(x_{s_i}|x_{L_{s_i}})$, and then, all of the densities $q(x_t|x_{L_{s_i}})$ are merged to form the new density $q(x_t|x_{L_t})$. Figure 4.4(b) provides a graphical illustration of the downward pass described in Algorithm 4.2 in conjunction with the recursive calculation proposed in Algorithm 4.1. These two steps can be considered separately or performed in concert with one another as demonstrated in Figure 4.4(b). Within the context of recursive estimation, the two-step process depicted in Figure 4.4(b) is more commonly called the *smoothing step* (*e.g.* see [13] for details) because it "smoothes" the estimates obtained in the upward pass by incorporating observations from other branches of the tree.

Using Algorithms 4.1 and 4.2, we now have the means to implement the iterations in (4.3) more efficiently than before. Specifically, at iteration i assume that the marginals $p^{(i)}(x_s, x_t)$ are known for each edge of the graph. Since $q^{(i)}$ is the projection of $p^{(i)}$ onto the graph $\mathcal{G}_{\preceq}$, the two marginals $q^{(i)}(x_s, x_t)$ and $p^{(i)}(x_s, x_t)$ are equal, and Algorithm 4.2 can then be used to determine the conditional densities $q^{(i)}(x_s|x_t, x_{L_s})$. Subsequently (or concurrently with Algorithm 4.2), Algorithm 4.1

[8] Recall from Definition 2.10 that a bottom-up ordering $(u_1, \dots, u_m)$ places the root vertex last. Furthermore, all child vertices of u_i must appear before vertex u_i in the ordering.

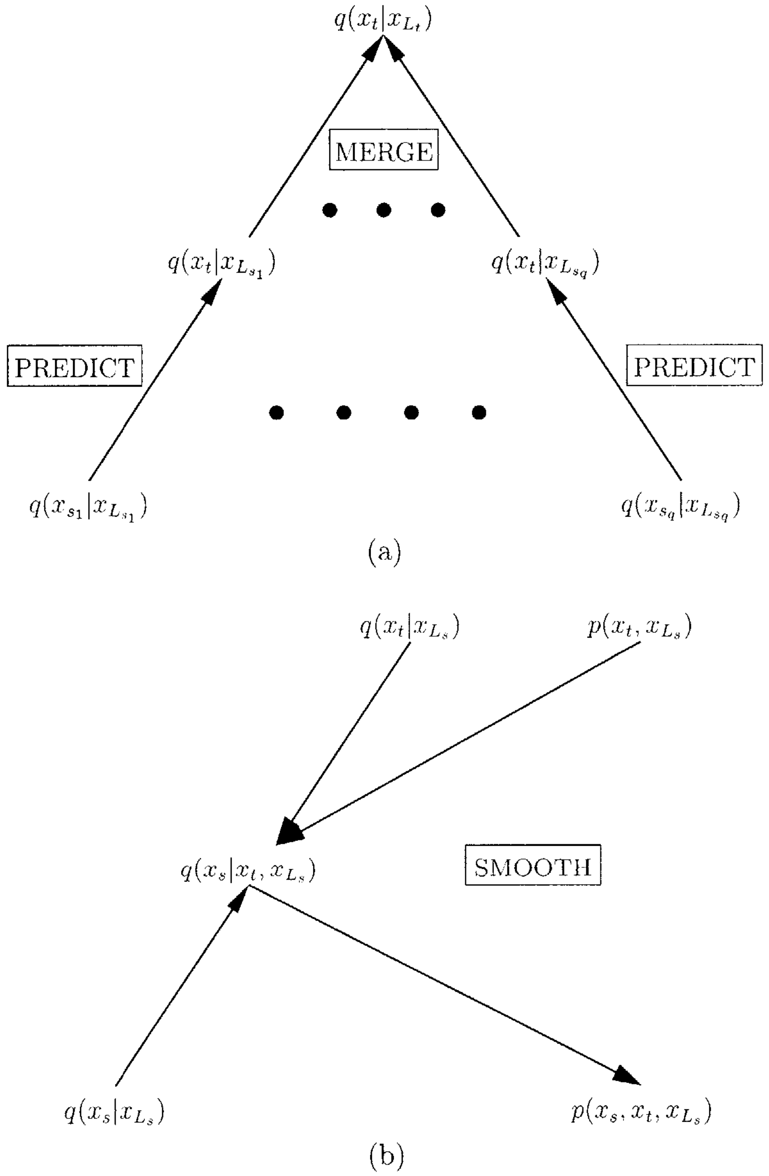


Figure 4.4. (a) Illustrates the prediction and merge steps which constitute the upward pass described in Algorithm 4.2. (b) Illustrates the smoothing step which constitutes the downward pass described in Algorithm 4.2 coupled with Algorithm 4.1.

can be used to calculate the marginals $p^{(i+1)}(x_s, x_t)$ along each edge of the tree. The iterations then continue in this same manner until the densities converge within some desired tolerance level. Section 4.4 uses this exact procedure to derive an algorithm specifically for the class of Gaussian multiscale models.

4.3 The EM Algorithm

In this section, we show how the iterative algorithm discussed thus far in this chapter is really the celebrated *expectation-maximization* (EM) algorithm in disguise. In order to make this comparison, we now recast the multiscale realization problem as a search over a parameterized space of densities, rather than the space of all densities; this reformulation is presented in Section 4.3.1. Section 4.3.2 then revisits the alternating minimization procedure originally discussed in Section 4.1.2 and shows how the iterations in (4.2) may be reformulated for the parameterized setting. Section 4.3.3 subsequently proves that the fixed points of this alternating procedure correspond to local extrema or saddle points with respect to the parametrization. Finally, Section 4.3.4 relates this parameterized form of the multiscale realization problem to the method of maximum-likelihood estimation and demonstrates that the proposed iterative procedure is a natural generalization of the EM algorithm.

4.3.1 Parameterized Densities

Consider now the approximate multiscale realization problem performed with respect to a parameterized set of densities. Specifically, let Θ be a specified set such that for all $\theta \in \Theta$, $q(x|\theta) \in \mathcal{P}_{\mathcal{G}_{\preceq}}(V, d)$ is a multiscale density. Rather than considering the space of all possible multiscale densities $q(x) \in \mathcal{P}_{\mathcal{G}_{\preceq}}(V, d)$, we focus on the proper subset $\{q(x|\theta)\}_{\theta \in \Theta}$. For a given Θ, we define the *parameterized approximate multiscale realization problem* $\widetilde{\mathcal{Q}}(\Theta)$ as follows:

> Parameterized Approximate Multiscale Realization Problem $\widetilde{\mathcal{Q}}(\Theta)$: Find any $\hat{\theta} \in \Theta$ which minimizes the cost $D\left(p^*(x_M) \| q(x_M|\hat{\theta})\right)$, *i.e.*
>
> $$\hat{\theta} = \arg\min_{\theta \in \Theta} D\left(p^*(x_M) \| q(x_M|\theta)\right).$$

In addition to problem $\widetilde{\mathcal{Q}}(\Theta)$, we also consider a related optimization problem which is indexed by a different space of densities. Let Γ be a specified set such that for all $\gamma \in \Gamma$, $p(x|\gamma) \in \mathcal{P}^M(V, d)$, *i.e.* each $p(x|\gamma)$ has the target marginal $p(x_M|\gamma) = p^*(x_M)$.[9] Rather than searching in the space of parameterized multiscale densities $\{q(x|\theta)\}_{\theta \in \Theta}$, we consider the problem of searching in the parameterized space $\{p(x|\gamma)\}_{\gamma \in \Gamma}$.

In order to define this additional optimization problem, it is necessary to link the two spaces of densities $\{p(x|\gamma)\}_{\gamma \in \Gamma}$ and $\{q(x|\theta)\}_{\theta \in \Theta}$, and this is accomplished by defining a special mapping from Γ to Θ. Specifically, let $\gamma \in \Gamma$ be fixed, and consider the following optimization problem,

$$\hat{\theta}_\gamma \triangleq \arg\min_{\theta \in \Theta} D\left(p(x|\gamma) \| q(x|\theta)\right). \tag{4.19}$$

[9] Unlike the density $q(x|\theta)$, $\theta \in \Theta$, which has the conditional independence structure of a multiscale model, the density $p(x|\gamma)$, $\gamma \in \Gamma$, does not necessarily have any special independence structure. However, each density $p(x|\gamma)$ is required to have the correct marginal $p^*(x_M)$.

If the value of $\hat{\theta}_\gamma$ exists and is unique for every $\gamma \in \Gamma$, then (4.19) defines a mapping from Γ to Θ and also implicitly a mapping from $\{p(x|\gamma)\}_{\gamma\in\Gamma}$ to $\{q(x|\theta)\}_{\theta\in\Theta}$. Although it is not essential that such a mapping exist between Γ and Θ, we henceforth assume that it does, since it allows us to make much stronger statements about the nature of optimization problem $\widetilde{\mathcal{Q}}(\Theta)$. In particular, the existence of this mapping (along with additional assumptions) allows us to state an important result about the convergence properties of the iterative procedure proposed in Section 4.3.2.

Using the decomposition in (3.55), the optimization problem in (4.19) may also be written as follows,

$$\begin{aligned}\hat{\theta}_\gamma &= \arg\min_{\theta\in\Theta} \left[D\left(p(x|\gamma)\|p^T(x|\gamma)\right) + D\left(p^T(x|\gamma)\|q(x|\theta)\right)\right] \\ &= \arg\min_{\theta\in\Theta} D\left(p^T(x|\gamma)\|q(x|\theta)\right). \qquad (4.20)\end{aligned}$$

The preceding problem makes it clear that $q(x|\hat{\theta}_\gamma)$ is the closest multiscale density to the projection $p^T(x|\gamma)$, and consequently, the mapping defined by (4.19) is a generalization of the tree projection mapping $p \longrightarrow p^T$ discussed in the previous chapter. To remind the reader of this fact, we denote the mapping defined by (4.19) as follows

$$\mathcal{T} : \Gamma \longrightarrow \Theta, \quad \mathcal{T}(\gamma) = \hat{\theta}_\gamma. \qquad (4.21)$$

Now, suppose that Γ and Θ are specified sets and that the mapping $\mathcal{T} : \Gamma \longrightarrow \Theta$ exists. Then, we define the *parameterized alternative approximate problem* $\widetilde{\mathcal{P}}^M(\Gamma)$ as follows:

Parameterized Alternative Approximate Problem $\widetilde{\mathcal{P}}^M(\Gamma)$: Find any $\hat{\gamma} \in \Gamma$ which minimizes the cost $D\left(p^*(x_M)\|q(x_M|\mathcal{T}(\hat{\gamma})\right)$, *i.e.*

$$\hat{\gamma} = \arg\min_{\gamma\in\Gamma} D\left(p^*(x_M)\|q(x_M|\mathcal{T}(\gamma)\right).$$

Notice that problem $\widetilde{\mathcal{P}}^M(\Gamma)$ is essentially identical to problem $\widetilde{\mathcal{Q}}(\Theta)$. In particular, both optimization problems use the same cost function $D\left(p^*(x_M)\|q(x_M|\theta)\right)$. The only difference between these two problems is the indexing set used for the search: problem $\widetilde{\mathcal{Q}}(\Theta)$ searches over Θ, while problem $\widetilde{\mathcal{P}}^M(\Gamma)$ searches over $\mathcal{T}(\Gamma)$, *i.e.* the image of Γ under the mapping $\mathcal{T}(\cdot)$.

The primary reason for considering problem $\widetilde{\mathcal{P}}^M(\Gamma)$ is that it offers additional degrees of freedom useful for solving problem $\widetilde{\mathcal{Q}}(\Theta)$. We discussed this same concept in Section 4.1.1 when considering alternative problem $\widetilde{\mathcal{P}}^M$, and in fact, problem $\widetilde{\mathcal{P}}^M(\Gamma)$ is simply the generalization of $\widetilde{\mathcal{P}}^M$ to a parameterized setting. Without additional constraints on the set Γ, however, problem $\widetilde{\mathcal{P}}^M(\Gamma)$ may or may not be a meaningful alternative to problem $\widetilde{\mathcal{Q}}(\Theta)$, or in other words, problems $\widetilde{\mathcal{P}}^M(\Gamma)$ and $\widetilde{\mathcal{Q}}(\Theta)$ may or may not be compatible.[10] The two problems are not compatible when the space of densities $\{p(x|\gamma)\}_{\gamma\in\Gamma}$ is not large enough to allow every solution $\hat{\theta}$ of problem $\widetilde{\mathcal{Q}}(\Theta)$ to have a corresponding solution $\hat{\gamma}$ of problem $\widetilde{\mathcal{P}}^M(\Gamma)$ with $\hat{\theta} = \mathcal{T}(\hat{\gamma})$.

To ensure that problems $\widetilde{\mathcal{P}}^M(\Gamma)$ and $\widetilde{\mathcal{Q}}(\Theta)$ are in fact compatible, we consider another mapping between the spaces Γ and Θ. Let $\theta \in \Theta$ be fixed, and consider the following optimization problem,

$$\hat{\gamma}_\theta \triangleq \arg\min_{\gamma\in\Gamma} D\left(p(x|\gamma)\|q(x|\theta)\right). \qquad (4.22)$$

[10] Recall from Section 3.4.1 that two problem formulations are compatible if there exists a surjective mapping from one solution space to the other solution space.

If the value of $\hat{\gamma}_\theta$ exists and is unique for every $\theta \in \Theta$, then (4.22) defines a mapping from Θ to Γ and also implicitly a mapping from $\{q(x|\theta)\}_{\theta \in \Theta}$ to $\{p(x|\gamma)\}_{\gamma \in \Gamma}$. Using the decomposition in (3.58), the optimization problem in (4.22) may also be written as follows,

$$\begin{aligned}\hat{\gamma}_\theta &= \arg\min_{\gamma \in \Gamma} \left[D\left(p(x|\gamma) \| \mathcal{F}^M(q(x|\theta))\right) + D\left(\mathcal{F}^M(q(x|\theta)) \| q(x|\theta)\right)\right] \\ &= \arg\min_{\gamma \in \Gamma} D\left(p(x|\gamma) \| \mathcal{F}^M(q(x|\theta))\right). \end{aligned} \tag{4.23}$$

The preceding problem makes it clear that $p(x|\hat{\gamma}_\theta)$ is the density closest to the projection $\mathcal{F}^M(q(x|\theta))$, and consequently, the mapping defined by (4.22) is a generalization of the projection mapping $q \longrightarrow \mathcal{F}^M(q)$ introduced in the previous chapter.

For our purposes, it is not sufficient to say that the mapping defined by (4.22) exists. In order to guarantee that problems $\widetilde{\mathcal{P}}^M(\Gamma)$ and $\widetilde{\mathcal{Q}}(\Theta)$ are compatible, the set Γ and the associated densities $\{p(x|\gamma)\}_{\gamma \in \Gamma}$ must be rich enough to allow the following condition to be satisfied:

$$\text{for every } \theta \in \Theta, \text{ there exists a } \gamma \in \Gamma \text{ such that } p(x|\gamma) = \mathcal{F}^M(q(x|\theta)).$$

In other words, for every $\theta \in \Theta$ we require the optimization problem in (4.23) to have a solution for which the KL divergence is zero. In fact, it is not necessary that this choice for $\gamma \in \Gamma$ be unique but only that one such choice exists. For convenience, we assume that the choice is unique, and we define the following mapping,

$$\mathcal{M} : \Theta \longrightarrow \Gamma, \quad \mathcal{M}(\theta) = \hat{\gamma}_\theta \text{ where } p(x|\hat{\gamma}_\theta) = \mathcal{F}^M(q(x|\theta)). \tag{4.24}$$

The notation $\mathcal{M}(\cdot)$ is used to remind the reader that this mapping is a generalization of the mapping $q \longrightarrow \mathcal{F}^M(q)$ to a parameterized setting.

If both mappings $\mathcal{T}(\cdot)$ and $\mathcal{M}(\cdot)$ exist for given sets Θ and Γ, then the solutions to problem $\widetilde{\mathcal{Q}}(\Theta)$ satisfy an important invariance property, as evidenced by the following proposition. This result is a generalization of the invariance property previously identified in Proposition 3.21 for solutions to problem $\widetilde{\mathcal{Q}}$.

Proposition 4.1 (An Invariance Property of Solutions to Problem $\widetilde{\mathcal{Q}}(\Theta)$).
Let $\mathcal{G}_\preceq$ be a rooted tree defined on vertex set V, and let $p^*(x_M)$ be a given target density. Let Θ and Γ be specified sets which index the densities $\{q(x|\theta)\}_{\theta \in \Theta} \subset \mathcal{P}_{\mathcal{G}_\preceq}(V,d)$ and $\{p(x|\gamma)\}_{\gamma \in \Gamma} \subset \mathcal{P}^M(V,d)$ respectively. If the mapping $\mathcal{T}(\cdot)$ in (4.21) and the mapping $\mathcal{M}(\cdot)$ in (4.24) both exist and if $\hat{\theta}$ is a solution to problem $\widetilde{\mathcal{Q}}(\Theta)$, then $\hat{\theta} = \mathcal{T}\left(\mathcal{M}\left(\hat{\theta}\right)\right)$.

Proof. See Appendix C.1. ■

If the assumptions stated in Proposition 4.1 hold, we can use the preceding invariance property to prove that problems $\widetilde{\mathcal{P}}^M(\Gamma)$ and $\widetilde{\mathcal{Q}}(\Theta)$ are compatible. Furthermore, any solution to problem $\widetilde{\mathcal{Q}}(\Theta)$ may be identified from some solution to problem $\widetilde{\mathcal{P}}^M(\Gamma)$ via the mapping $\mathcal{T}(\cdot)$.

Proposition 4.2 (Relationship Between Solutions to $\widetilde{\mathcal{P}}^M(\Gamma)$ and $\widetilde{\mathcal{Q}}(\Theta)$).
Suppose the assumptions stated in Proposition 4.1 hold; in particular, suppose the mappings $\mathcal{T}(\cdot)$ and $\mathcal{M}(\cdot)$ exist. Then, the mapping $\mathcal{T}(\cdot)$ is a surjection from the solution set of problem $\widetilde{\mathcal{P}}^M(\Gamma)$ onto the solution set of problem $\widetilde{\mathcal{Q}}(\Theta)$.

Proof. See Appendix C.2. ■

Proposition 4.2 demonstrates that solutions to problem $\widetilde{\mathcal{Q}}(\Theta)$ can be obtained from solutions to alternative problem $\widetilde{\mathcal{P}}^M(\Gamma)$ if the sets Θ and Γ are appropriately chosen. For the non-parameterized problems considered in the preceding chapter, Proposition 3.22 analogously proves that problems $\widetilde{\mathcal{P}}^M$ and $\widetilde{\mathcal{Q}}$ are compatible. In this latter case, the two mappings $p \longrightarrow p^T$ and $q \longrightarrow \mathcal{F}^M(q)$ are used to show that a surjection exists between the two solution sets, and since these mappings are always defined on the space of all densities, such a surjection is guaranteed to exist. As Proposition 4.2 demonstrates, though, we must be more careful when considering parameterized spaces if we want to take advantage of the additional degrees of freedom afforded by alternative problem $\widetilde{\mathcal{P}}^M(\Gamma)$. As the following section demonstrates, if the assumptions in Proposition 4.1 are satisfied, we can in theory implement (within a parameterized setting) the iterative algorithm discussed in Section 4.1.

■ 4.3.2 Alternating Minimizations in a Parameterized Space

Using the ideas introduced in the previous section, this section presents an iterative approach for solving the parameterized approximate multiscale realization problem $\widetilde{\mathcal{Q}}(\Theta)$. This algorithm is similar to the iterative approach suggested in Section 4.1 for solving problem $\widetilde{\mathcal{Q}}$, but rather than performing the iterations in the space of all densities, we now restrict our attention to the two subspaces $\{p(x|\gamma)\}_{\gamma\in\Gamma}$ and $\{q(x|\theta)\}_{\theta\in\Theta}$. Specifically, given an initial guess for $\theta^{(0)}$, we consider the following sequence of alternating minimizations for $i = 1, 2, \ldots,$

$$\gamma^{(i)} \triangleq \arg\min_{\gamma\in\Gamma} D\left(p(x|\gamma)\|q(x|\theta^{(i-1)})\right), \tag{4.25a}$$

$$\theta^{(i)} \triangleq \arg\min_{\theta\in\Theta} D\left(p(x|\gamma^{(i)})\|q(x|\theta)\right). \tag{4.25b}$$

Intuitively, the alternating minimizations in (4.25) attempt to minimize the cost function $\mathcal{C}(\gamma,\theta) \triangleq D\left(p(x|\gamma)\|q(x|\theta)\right)$ over all $\gamma \in \Gamma$ and $\theta \in \Theta$. From this perspective, we see that the iterations are simply a method of coordinate descent: (4.25a) minimizes the cost with respect to the first set of coordinates $\gamma \in \Gamma$, while (4.25b) minimizes the cost with respect to the second set of coordinates $\theta \in \Theta$. Using the fact that $\gamma^{(i)}$ and $\theta^{(i)}$ are minimizers of their respective optimization problems, the following sequence of inequalities holds,

$$D\left(p(x|\gamma^{(i+1)})\|q(x|\theta^{(i+1)})\right) \leq D\left(p(x|\gamma^{(i+1)})\|q(x|\theta^{(i)})\right) \leq D\left(p(x|\gamma^{(i)})\|q(x|\theta^{(i)})\right),$$

and therefore, the iterations in (4.25) decrease (or at least do not increase) the cost $\mathcal{C}(\gamma,\theta)$.

From a slightly different perspective, the iterations in (4.25) attempt to find a density $p(x|\gamma)$ with the correct marginal $p^*(x_M)$ which is closest to a multiscale density $q(x|\theta)$. If there exists a $\gamma \in \Gamma$ and $\theta \in \Theta$ such that $p(x|\gamma) = q(x|\theta)$ then an exact solution to problem $\widetilde{\mathcal{Q}}(\Theta)$ can be found; otherwise, the iterations in (4.25) attempt to find an appropriate tradeoff. However, it is important to note that the objective of problem $\widetilde{\mathcal{Q}}(\Theta)$ is not to find two densities $p(x|\gamma)$ and $q(x|\theta)$ defined on the full space $\mathcal{X}$ which are equal (or approximately equal) to each other. Rather, the goal is find a multiscale density $q(x|\theta)$ such that the marginal $q(x_M|\theta)$ matches the target density $p^*(x_M)$. This suggests that the alternating minimizations in (4.25) do not necessarily generate solutions to problem $\widetilde{\mathcal{Q}}(\Theta)$. By imposing certain constraints on the sets Γ and Θ, however, the iterates in (4.25) are in fact guaranteed to converge to fixed points $\hat{\theta}$ which satisfy the necessary conditions for optimality stated in Proposition 4.1.

Consider again the assumptions stated in Proposition 4.1 – the constraints needed for problems $\widetilde{\mathcal{Q}}(\Theta)$ and $\widetilde{\mathcal{P}}^M(\Gamma)$ to be compatible. Specifically, assume that the mappings $\mathcal{T}(\cdot)$ and $\mathcal{M}(\cdot)$ both exist. If this is the case, the iterations in (4.25) may equivalently be written as follows,[11]

$$\gamma^{(i)} = \mathcal{M}\left(\theta^{(i-1)}\right), \tag{4.26a}$$

$$\theta^{(i)} = \mathcal{T}\left(\gamma^{(i)}\right). \tag{4.26b}$$

In other words, (4.26a) indicates that there exists a $\gamma^{(i)} \in \Gamma$ such that $p(x|\gamma^{(i)}) = \mathcal{F}^M(q(x|\theta^{(i-1)}))$ is the minimizer of (4.25a), and (4.26b) indicates that there exists a unique choice for $\theta^{(i)}$ which minimizes (4.25b).

Assuming for the moment that the iterations in (4.26) converge, notice that all fixed points must satisfy $\hat{\theta} = \mathcal{T}\left(\mathcal{M}\left(\hat{\theta}\right)\right)$. As Proposition 4.1 indicates, the solutions to problem $\widetilde{\mathcal{Q}}(\Theta)$ must also satisfy this same fixed-point equation. Consequently, all fixed points of the iterations in (4.26) satisfy the necessary conditions for optimality.

Consider now the issue of whether or not the sequence $\{\theta^{(i)}\}$ generated by (4.26) converges to a fixed point $\hat{\theta}$. To aid in this effort, we examine the sequence of non-negative real numbers ε_i defined as follows for $i = 0, 1, 2, \ldots$,

$$\varepsilon_i \triangleq D\left(p^*(x_M)\|q(x_M|\theta^{(i)})\right). \tag{4.27}$$

Using (4.1), the value of ε_i may also be written as follows,

$$\varepsilon_i = D\left(p(x|\gamma^{(i+1)})\|q(x|\theta^{(i)})\right) - D\left(p(x|\gamma^{(i+1)})\|\mathcal{F}^M\left(q(x|\theta^{(i)})\right)\right) = D\left(p(x|\gamma^{(i+1)})\|q(x|\theta^{(i)})\right),$$

where the choice $\gamma^{(i+1)} = \mathcal{M}(\theta^{(i)})$ implies that $p(x|\gamma^{(i+1)}) = \mathcal{F}^M(q(x|\theta^{(i)}))$, thereby setting the second term to zero. The minimization problem in (4.25b) then implies the following inequality,

$$\varepsilon_i = D\left(p(x|\gamma^{(i+1)})\|q(x|\theta^{(i)})\right) \geq D\left(p(x|\gamma^{(i+1)})\|q(x|\theta^{(i+1)})\right), \tag{4.28}$$

and if the value of $\theta^{(i+1)}$ is unique in (4.25b), *i.e.* $\theta^{(i+1)} = \mathcal{T}(\gamma^{(i+1)})$, the inequality in (4.28) holds with equality if and only if $\theta^{(i)} = \theta^{(i+1)}$.

The expression on the right-hand-side of (4.28) may be further decomposed using (4.1),

$$\begin{aligned} D\left(p(x|\gamma^{(i+1)})\|q(x|\theta^{(i+1)})\right) &= D\left(p(x|\gamma^{(i+1)})\|\mathcal{F}^M\left(q(x|\theta^{(i+1)})\right)\right) + D\left(p^*(x_M)\|q(x_M|\theta^{(i+1)})\right) \\ &= D\left(p(x|\gamma^{(i+1)})\|p(x|\gamma^{(i+2)})\right) + \varepsilon_{i+1}. \end{aligned} \tag{4.29}$$

Combining (4.28) and (4.29) gives

$$\varepsilon_{i+1} \leq \varepsilon_i - D\left(p(x|\gamma^{(i+1)})\|p(x|\gamma^{(i+2)})\right). \tag{4.30}$$

This inequality proves that the iterations in (4.26) generate a non-increasing sequence of real numbers $\{\varepsilon_i\}$ which is bounded below by zero. Therefore, the sequence must converge to some value $\bar{\varepsilon} \geq 0$ [91]. Of course, we are more interested in the properties of the sequence $\{\theta^{(i)}\}$ than in $\{\varepsilon_i\}$. As the following proposition states, the existence of the mappings $\mathcal{T}(\cdot)$ and $\mathcal{M}(\cdot)$, along with additional conditions on the set Θ, guarantee that $\{\theta^{(i)}\}$ converges to a fixed point in the limit.

[11] Recall that the two minimization problems in (4.25) were previously considered in Section 4.3.1 when defining the mappings $\mathcal{T}(\cdot)$ and $\mathcal{M}(\cdot)$.

Proposition 4.3 (Convergence of the Sequence $\{\theta^{(i)}\}$).

Suppose the assumptions stated in Proposition 4.1 hold; in particular, suppose the mappings $\mathcal{T}(\cdot)$ and $\mathcal{M}(\cdot)$ exist. Let Θ be a subset of some metric space. Given an initial starting point $\theta^{(0)}$ such that $\varepsilon_0 < \infty$, let Θ_0 denote the subset of all $\theta \in \Theta$ satisfying $D\left(p^*(x_M) \| q(x_M|\theta)\right) \leq D\left(p^*(x_M) \| q(x_M|\theta^{(0)})\right)$, i.e.

$$\Theta_0 \triangleq \left\{\theta \in \Theta \,\middle|\, D\left(p^*(x_M) \| q(x_M|\theta)\right) \leq D\left(p^*(x_M) \| q(x_M|\theta^{(0)})\right)\right\}. \tag{4.31}$$

If Θ_0 is a compact subset of Θ, then the sequence $\{\theta^{(i)}\}$ generated by (4.26) converges to a fixed point $\hat{\theta}$ which satisfies $\hat{\theta} = \mathcal{T}(\mathcal{M}(\hat{\theta}))$.

Proof. Since the sequence $\{\varepsilon_i\}$ is non-increasing, each $\theta^{(i)}$ lies in the set Θ_0, and consequently, $\{\theta^{(i)}\}$ is a sequence in a compact metric space. Then, some subsequence of $\{\theta^{(i)}\}$ must converge in the limit [91].

Since the sequence $\{\varepsilon_i\}$ also converges, the inequality in (4.30) must be an equality in the limit, and the second term on the right-hand-side of (4.30) must approach zero. We assume here that the mapping $\mathcal{T}(\cdot)$ exists; consequently the value of $\theta^{(i+1)} = \mathcal{T}(\gamma^{(i+1)})$ in (4.28) is unique. This means that the inequality in (4.30) can only be an equality at the point of convergence. Therefore, the sequence $\{\theta^{(i)}\}$ must converge to a fixed point satisfying $\hat{\theta} = \mathcal{T}(\mathcal{M}(\hat{\theta}))$. ■

To summarize the preceding discussion, the two mappings $\mathcal{T}(\cdot)$ and $\mathcal{M}(\cdot)$ are important tools for establishing the convergence of $\{\theta^{(i)}\}$. The mapping $\mathcal{M}(\cdot)$ must exist in order for the sequence $\{\varepsilon_i\}$ to be non-increasing; without the existence of this mapping, the alternating minimizations in (4.25), while attempting to minimize an upper bound, do not necessarily minimize the cost function $D\left(p^*(x_M) \| q(x_M|\theta)\right)$ of interest. If the mapping $\mathcal{T}(\cdot)$ exists, then as demonstrated in the proof of Proposition 4.3, it is straightforward to show that $\{\theta^{(i)}\}$ converges to a unique fixed point in the limit.

Requiring the mapping $\mathcal{T}(\cdot)$ to exist is a rather stringent condition for some parameterized problems, and as a result, similar conditions, which also guarantee convergence, have been considered in the literature [78, 110] but discussed within the context of the EM algorithm. As an example, the conditions cited in [110] require the set Θ_0 to be compact and the mapping $\mathcal{M}(\cdot)$ to exist, just as we have assumed in Proposition 4.3; however, these conditions do not assume the existence of the mapping $\mathcal{T}(\cdot)$. For problems where Θ_0 is compact and $\mathcal{M}(\cdot)$ exists, the EM iterations are guaranteed to converge, but in the limit, the iterates may cycle between a set of limit points. This is not an issue in a practical sense, since each of these limit points is an equally viable solution to the realization problem.

For our purposes, the conditions stated in Proposition 4.3 are easily verifiable for the Gaussian problem considered in Section 4.4, and as such, we continue to focus on them. As we soon discuss, both mappings $\mathcal{M}(\cdot)$ and $\mathcal{T}(\cdot)$ exist in the Gaussian realization problem, thereby ensuring the monotonic properties of the sequence $\{\varepsilon_i\}$. In addition, since the class of Gaussian multiscale models has a finite-dimensional parametrization, the set Θ_0 is compact if and only if it is both closed and bounded. As subsequent discussion reveals, the set Θ_0 is unbounded for every initial starting point in the Gaussian problem, and as such, additional constraints are imposed to deal with this issue.

■ 4.3.3 Parameterizations and Local Minima

Proposition 4.3 states conditions under which the sequence $\{\theta^{(i)}\}$ generated by (4.26) is guaranteed to converge to a fixed point $\hat{\theta} = \mathcal{T}\left(\mathcal{M}\left(\hat{\theta}\right)\right)$, and Proposition 4.1 indicates that all solutions to problem $\tilde{\mathcal{Q}}(\Theta)$ satisfy this same fixed-point equation. In this section, we examine these fixed points in more detail, and in particular, we show that the gradient of the function $D\left(p^*(x_M)\|q(x_M|\theta)\right)$ vanishes at each of these fixed points. Consequently, each such $\hat{\theta}$ is either a local extrema or saddle point of the optimization problem $\tilde{\mathcal{Q}}(\Theta)$.

To see this, assume the function $\mathcal{C}(\theta) \triangleq D\left(p^*(x_M)\|q(x_M|\theta)\right)$ is differentiable on the interior of the set Θ. Then, using the decomposition in (4.1), we can write the gradient of $\mathcal{C}(\theta)$ as follows,

$$\frac{\partial}{\partial\theta}\mathcal{C}(\theta) = \frac{\partial}{\partial\theta}\left[D\left(p(x|\gamma)\|q(x|\theta)\right) - D\left(p(x|\gamma)\|\mathcal{F}^M(q(x|\theta))\right)\right]. \tag{4.32}$$

Since (4.32) holds for any value of $\gamma \in \Gamma$, it holds for the value $\gamma = \mathcal{M}(\theta)$. For this choice of γ, the second term in (4.32) is zero, thereby giving

$$\frac{\partial}{\partial\theta}\mathcal{C}(\theta) = \frac{\partial}{\partial\theta}D\left(p(x|\gamma)\|q(x|\theta)\right). \tag{4.33}$$

Consider now any fixed point $\hat{\theta} = \mathcal{T}(\mathcal{M}(\hat{\theta}))$, and assume that $\hat{\theta}$ lies in the interior of the set Θ. Defining $\hat{\gamma} \triangleq \mathcal{M}(\hat{\theta})$ and using (4.33), the gradient of $\mathcal{C}(\theta)$ at the point $\theta = \hat{\theta}$ is given by,

$$\left.\frac{\partial}{\partial\theta}\mathcal{C}(\theta)\right|_{\theta=\hat{\theta}} = \left.\left[\frac{\partial}{\partial\theta}D\left(p(x|\hat{\gamma})\|q(x|\theta)\right)\right]\right|_{\theta=\hat{\theta}}. \tag{4.34}$$

Now, since the fixed point $\hat{\theta}$ satisfies $\hat{\theta} = \mathcal{T}(\hat{\gamma})$ and therefore is the unique minimizer of the cost $D\left(p(x|\hat{\gamma})\|q(x|\theta)\right)$ in (4.25b), the gradient $\frac{\partial}{\partial\theta}D\left(p(x|\hat{\gamma})\|q(x|\theta)\right)$ must be zero at $\theta = \hat{\theta}$. Then, according to (4.34), the gradient of $\mathcal{C}(\theta)$ at $\theta = \hat{\theta}$ must also be zero. This fact is summarized in the following proposition.

Proposition 4.4 (Fixed Points are Points of Zero Gradient).
Suppose the assumptions stated in Proposition 4.1 hold; in particular, suppose the mappings $\mathcal{T}(\cdot)$ and $\mathcal{M}(\cdot)$ exist. Suppose $\hat{\theta} = \mathcal{T}\left(\mathcal{M}\left(\hat{\theta}\right)\right)$ is a fixed point of the iterations in (4.26) and lies in the interior of the set Θ. If the function $D\left(p^*(x_M)\|q(x|\theta)\right)$ is differentiable at the point $\theta = \hat{\theta}$, then the gradient is zero at this point.

Proof. See the preceding discussion. ■

Proposition 4.4 demonstrates an important property possessed by the fixed points of the iterations in (4.26). Not only is the fixed-point equation $\hat{\theta} = \mathcal{T}(\mathcal{M}(\hat{\theta}))$ a necessary condition for solutions to problem $\tilde{\mathcal{Q}}(\Theta)$, as evidenced by Proposition 4.1, but any point satisfying this equation also satisfies the first-order necessary conditions for optimality. As a result, such a point may be a local minimum, local maximum, or a saddle point. In general, the probability of finding a local maximum is zero, since the iterations would need to be initialized to this point. Therefore, the iterations in (4.26) tend to converge to either local minima or saddle points. The following example is interesting because it demonstrates a simple problem where both of these points exist, and in addition, it provides a preview of the Gaussian realization problem discussed in Section 4.4.

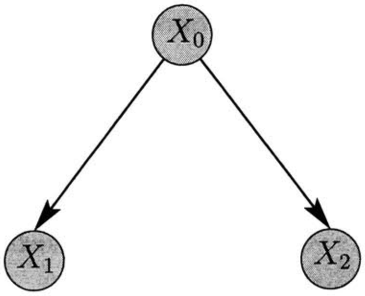


Figure 4.5. Multiscale model considered in Example 4.2, where X_0, X_1, X_2 are all scalar random variables and $M = \{1, 2\}$.

Example 4.2 (Local Minima and Saddle Points).
This example examines the multiscale realization problem for the model shown in Figure 4.5 and where a target density $p^*(x_M)$ is specified for the leaf vertices $M = \{1, 2\}$. In this example, X_1 and X_2 are jointly Gaussian zero-mean scalar random variables (under the density $p^*(x_M)$) with the following covariance matrix

$$E\left[\left[\begin{array}{c} X_1 \\ X_2 \end{array}\right]\left[\begin{array}{cc} X_1 & X_2 \end{array}\right]\right] = \left[\begin{array}{cc} 1 & \rho_{x_1 x_2} \\ \rho_{x_1 x_2} & 1 \end{array}\right]. \tag{4.35}$$

The goal is to specify a scalar random variable X_0 such that X_1 and X_2 are conditionally independent given X_0. Of course, this problem is trivial in the sense that $X_0 = X_1$ and $X_0 = X_2$ are valid solutions which generate an exact multiscale model. Nonetheless, this problem still provides valuable insight into the Gaussian realization problem.

In this example, we constrain the set of multiscale densities $\{q(x_0, x_1, x_2|\theta)\}_{\theta \in \Theta}$ to be zero-mean and Gaussian, and therefore, the goal is to minimize the cost $D\left(p^*(x_1, x_2) \| q(x_1, x_2|\theta)\right)$ over this set. Using the covariance in (4.35), it can be shown that this cost may be written as follows,

$$D\left(p^*(x_1, x_2) \| q(x_1, x_2|\theta)\right) = \frac{1}{2} \log \left[\frac{1 - \rho_{x_1 x_0}^2 \rho_{x_2 x_0}^2}{1 - \rho_{x_1 x_2}^2}\right] + \left[\frac{1 - \rho_{x_1 x_0} \rho_{x_2 x_0} \rho_{x_1 x_2}}{1 - \rho_{x_1 x_0}^2 \rho_{x_2 x_0}^2}\right] - 1, \tag{4.36}$$

where $\rho_{x_i x_j}$ represents a correlation between X_i and X_j. In (4.36), $\rho_{x_1 x_2}$ is the correlation between X_1 and X_2 under the target density $p^*(x_M)$ as specified in the covariance matrix in (4.35). On the other hand, $\rho_{x_1 x_0}$ and $\rho_{x_2 x_0}$ are parameters of the multiscale model which define the correlations between X_1 and X_0 (and X_2 and X_0) with respect to the multiscale density $q(x|\theta)$.[12]

The surface in Figure 4.6(a) shows the cost in (4.36) plotted as a function of the two parameters $\rho_{x_1 x_0}$ and $\rho_{x_2 x_0}$, with the choice $\rho_{x_1 x_2} = 0.25$. In this example, the surface is extremely flat, making it difficult to distinguish several important features of the problem; as a visual aid, the function $\log_{10}\left[0.01 + D\left(p^*(x_1, x_2) \| q(x_1, x_2|\theta)\right)\right]$ is plotted in Figure 4.6(b). This plot makes it clear that the cost in (4.36) has a saddle point at $(0, 0)$ and an infinite number of local minima which also correspond to global minima. These local minima are located in two separate "troughs" with endpoints at $(1, 0.25)$, $(0.25, 1)$, $(-1, -0.25)$, and $(-0.25, -1)$ – endpoints which correspond to the trivial solutions $X_0 = X_1$, $X_0 = X_2$, $X_0 = -X_1$, and $X_0 = -X_2$ respectively.

[12] In this particular example, the cost in (4.36) is not a function of the variance of X_0.

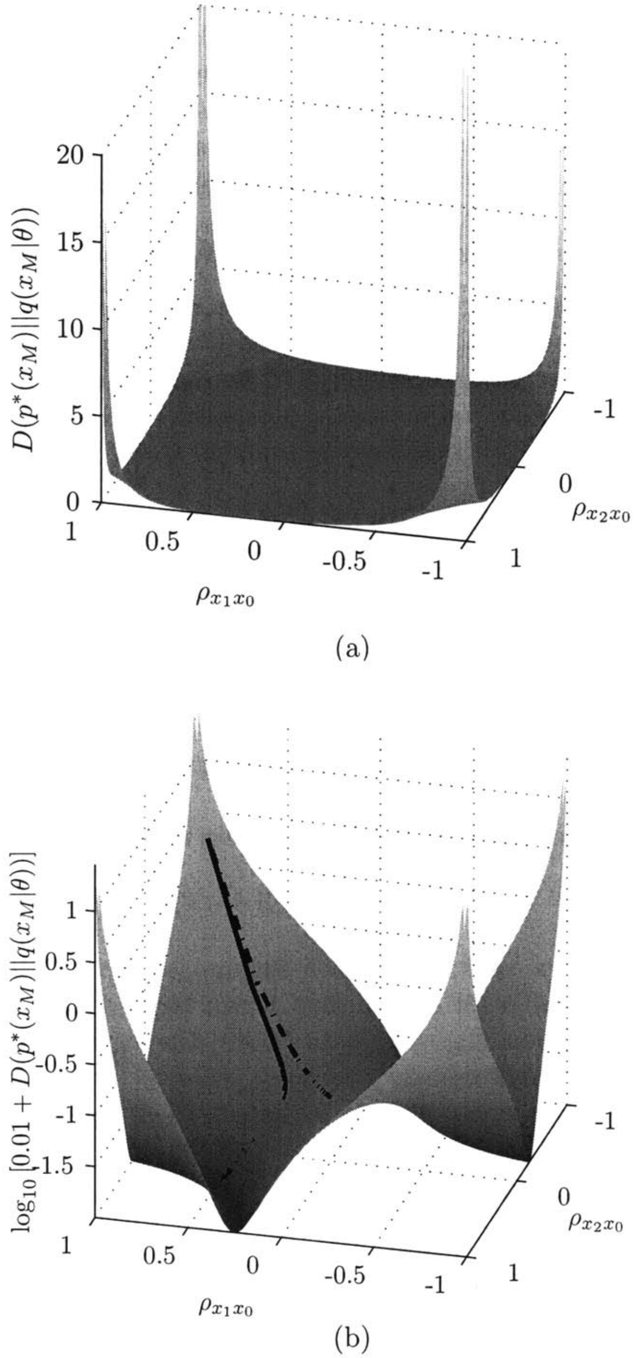


Figure 4.6. (a) Plot of the KL divergence $D\left(p^*(x_M)\|q(x_M|\theta)\right)$ for the Gaussian multiscale realization problem considered in Example 4.2. The plot shows the KL divergence as a function of the two correlations $\rho_{x_1x_0}$ and $\rho_{x_2x_0}$. (b) Same plot as in (a) but on a different scale, namely $\log_{10}\left[0.01 + D\left(p^*(x_M)\|q(x_M|\theta)\right)\right]$. The plot shows that there exists one saddle point at $(0, 0)$ and an infinite number of local minima that are also global minima. The dashed line shows the path of the EM algorithm given an initial starting point of $(0.9, -0.9)$; in this case, the algorithm converges to the saddle point. The solid line shows the path of the EM algorithm given an initial starting point of $(0.905, -0.9)$.

The plot in Figure 4.6(b) also shows the trajectory of the iterations in (4.26) for two different starting points. When the starting point $(0.9, -0.9)$ is chosen, the iterates follow the dashed line and converge to the saddle point at $(0, 0)$. In fact, any starting point of the form $(a, -a)$, $-1 < a < 1$, generates a sequence which converges to this saddle point. For all other starting points, the sequence converges to a global minimum. For example, the starting point $(0.905, -0.9)$ generates the trajectory represented by the solid line in Figure 4.6(b). ◀

■ 4.3.4 Maximum-Likelihood Estimation and the EM Algorithm

In this section, we take a brief aside to relate the parameterized approximate multiscale realization problem $\tilde{Q}(\Theta)$, introduced in Section 4.3.1, to *maximum-likelihood (ML) estimation* [26], and we relate the iterative procedure discussed in Section 4.3.2 to the EM algorithm [27]. First, the relationship between maximum-likelihood estimation and problem $\tilde{Q}(\Theta)$ can be seen by decomposing the cost function $D\left(p^*(x_M) \| q(x_M|\theta)\right)$ considered in problem $\tilde{Q}(\Theta)$. Specifically, we can write[13]

$$\begin{aligned} D\left(p^*(x_M) \| q(x_M|\theta)\right) &= \int p^*(x_M) \log\left(\frac{p^*(x_M)}{q(x_M|\theta)}\right) dx_M \\ &= \int p^*(x_M) \log p^*(x_M) dx_M - \int p^*(x_M) \log q(x_M|\theta) dx_M \\ &= C - E_{p^*}\left[\log q(X_M|\theta)\right]. \end{aligned} \tag{4.37}$$

As (4.37) demonstrates, the first term involves only the target density $p^*(x_M)$ and is therefore a constant with respect to θ, while the second term is the negative expectation of $\log q(X_M|\theta)$ under the target density $p^*(x_M)$.

Using (4.37), we can equivalently state problem $\tilde{Q}(\Theta)$ as the following maximization problem,

$$\hat{\theta} = \arg\max_{\theta \in \Theta} E_{p^*}\left[\log q(X_M|\theta)\right], \tag{4.38}$$

or in other words, find the value of $\theta \in \Theta$ which maximizes the *expected log-likelihood*. This is a generalization of the method of ML estimation which seeks to maximize the log-likelihood $\log q(X_M|\theta)$ for a single observation $X_M = x_M$. In (4.38), the log-likelihood is averaged over all possible observations using the target density $p^*(x_M)$.

If we consider the special case where $p^*(x_M)$ is the empirical density for a set of N independent observations $x_M^{(1)}, x_M^{(2)}, \ldots, x_M^{(N)}$, *i.e.* $p^*(x_M) = \frac{1}{N}\sum_{i=1}^{N} \delta(x_M - x_M^{(i)})$, then the criterion to be maximized in (4.38) can be rewritten as follows,

$$\begin{aligned} E_{p^*}\left[\log q(X_M|\theta)\right] = \int p^*(x_M) \log q(x_M|\theta) dx_M &= \int \left[\frac{1}{N}\sum_{i=1}^{N} \delta(x_M - x_M^{(i)})\right] \log q(x_M|\theta) dx_M \\ &= \frac{1}{N}\sum_{i=1}^{N} \log q(x_M^{(i)}|\theta) = \frac{1}{N} \log\left[\prod_{i=1}^{N} q(x_M^{(i)}|\theta)\right]. \end{aligned} \tag{4.39}$$

The expression in (4.39) is proportional to the log-likelihood, under the assumption of independent observations. Therefore, when the goal is to find a multiscale model that "best" fits N independent

[13] In some cases, the KL divergence on the left-hand-side of (4.37) may be finite while one of the terms on the right-hand-side of (4.37) is infinite. For the purpose of drawing a parallel to ML estimation, we assume that a target density $p^*(x_M)$ has been specified such that this is not the case.

observations $x_M^{(1)}, x_M^{(2)}, \ldots, x_M^{(N)}$, problem $\widetilde{Q}(\Theta)$ is equivalent to ML estimation with respect to a parameterized set of densities $\{q(x|\theta)\}_{\theta \in \Theta}$.

In general, ML estimation problems, such as the one considered in (4.38), can be difficult to solve due to the existence of multiple local maxima. Since problem $\widetilde{Q}(\Theta)$ is equivalent to (4.38), it may possess multiple local minima (depending on the chosen parametrization and the target density $p^*(x_M)$), as discussed in the previous section. It is well-known that the EM algorithm can be useful in attempting to solve some ML estimation problems, specifically problems where there are "missing" data. In (4.38), we have this problem of missing data because only the variables X_M are defined (by the target density $p^*(x_M)$) – the remaining variables X_{V-M} are unknown or missing.

As we now discuss, the iterative algorithm introduced in Section 4.3.2 is a generalization of the EM algorithm and essentially provides a method for probabilistically characterizing the unknown variables X_{V-M}. Specifically, the mapping $\mathcal{M}(\cdot)$ provides the necessary characterization. To see this, suppose $\gamma^{(i)} = \mathcal{M}(\theta^{(i-1)})$ as in (4.26a); then, by definition the density $p(x|\gamma^{(i)})$ satisfies

$$p(x|\gamma^{(i)}) = \mathcal{F}^M\left(q(x|\theta^{(i-1)})\right) = q(x_{V-M}|x_M, \theta^{(i-1)})p^*(x_M).$$

The density $p(x|\gamma^{(i)})$ is a so-called *complete density* since it is defined on the complete set of variables X_V. This complete density maintains the same marginal density $p^*(x_M)$ on the variables X_M and defines the unknown variables X_{V-M} using the conditional density $q(x_{V-M}|x_M, \theta^{(i-1)})$. This conditional density provides the best estimate for X_{V-M} conditioned on X_M under the current model $q(x|\theta^{(i-1)})$.

Given the complete density $p(x|\gamma^{(i)})$, consider now the optimization problem in (4.25b). The cost function $D\left(p(x|\gamma^{(i)})\|q(x|\theta)\right)$ can be decomposed in the same manner in which the cost $D\left(p^*(x_M)\|q(x_M|\theta)\right)$ was decomposed in (4.37). Doing so gives the following,

$$D\left(p(x|\gamma^{(i)})\|q(x|\theta)\right) = C - E_{p(x|\gamma^{(i)})}\left[\log q(X|\theta)\right], \tag{4.40}$$

where C is a constant with respect to θ. Using (4.40), the optimization problem in (4.25b) can equivalently be posed as follows,

$$\theta^{(i)} = \arg\max_{\theta \in \Theta} E_{p(x|\gamma^{(i)})}\left[\log q(X|\theta)\right], \tag{4.41}$$

which is a maximum-likelihood estimation problem where the averaging density is the complete density $p(x|\gamma^{(i)})$.

Based on the preceding discussion, the alternating minimization problems in (4.25) can equivalently be stated as follows (assuming that the mapping $\mathcal{M}(\cdot)$ exists):

$$\begin{aligned} &\underline{\text{E-step:}} \quad p(x|\gamma^{(i)}) = \mathcal{F}^M\left(q(x|\theta^{(i-1)})\right) = q(x_{V-M}|x_M, \theta^{(i-1)})p^*(x_M), \\ &\underline{\text{M-step:}} \quad \theta^{(i)} = \arg\max_{\theta \in \Theta} E_{p(x|\gamma^{(i)})}\left[\log q(X|\theta)\right]. \end{aligned}$$

The first step, the so-called expectation step of the EM algorithm, generates the averaging density $p(x|\gamma^{(i)})$, while the second step, the so-called maximization step of the EM algorithm, finds a value of $\theta \in \Theta$ which maximizes the expected log-likelihood.

The relationship between the EM algorithm and the problem of alternating optimizations has been noted by several authors including [20, 59, 81]. In [81], alternating maximizations are performed with respect to the variational free energy from statistical physics, and these alternating maximizations are shown to be equivalent to the iterations of EM. In [20], the alternating minimization problems are identical to the ones considered here, except the search sets for these problems are considered more generically and are not specifically sets of densities which possess conditional independence or marginal constraints. As we have demonstrated here, though, these search sets must satisfy certain constraints in order for the alternating minimizations in (4.25) to generate solutions to problem $\tilde{\mathcal{Q}}(\Theta)$.

4.4 Realizing Gaussian Multiscale Models Given Exact Statistics

Using the ideas presented in the previous three sections, we are now in a position to derive an algorithm for finding a solution to the Gaussian multiscale realization problem. Section 4.4.1 describes the specific problem to be solved and shows how the convergence results presented in Section 4.3 may be utilized in the problem of interest here. Section 4.4.2 then uses the algorithm described in Section 4.2 to derive an efficient algorithm for solving the Gaussian realization problem, and Section 4.4.3 suggests a re-scaled version of this algorithm which ensures that the iterations always converge to a fixed point. Finally, Section 4.4.4 provides several illustrative examples which demonstrate the convergence properties of the algorithm in practice.

4.4.1 The Problem Setup

Consider the multiscale realization problem where the target density $p^*(x_M)$ is Gaussian with zero mean and covariance P_M^*, *i.e.*[14]

$$p^*(x_M) = N\left(x_M; 0, P_M^*\right).$$

We henceforth assume that P_M^* is positive definite and therefore invertible. Given such a target density, our goal is to solve problem $\tilde{\mathcal{Q}}(\Theta)$ with respect to the set of multiscale densities $\{q(x|\theta)\}_{\theta\in\Theta}$, where each $q(x|\theta) = N\left(x; 0, Q^\theta\right)$ is a zero-mean Gaussian with positive-definite covariance matrix Q^θ.

As we now discuss, this particular realization problem may be viewed as a structured matrix optimization problem. Specifically, since each $q(x|\theta)$, $\theta \in \Theta$, is a multiscale density and therefore has special factorization structure, the covariance matrix Q^θ associated with $q(x|\theta)$ has special structure as well [71, 97]. To see this structure, let the set B_{uv} contain the indices of the matrix Q^θ which correspond to the cross-covariance between random vectors X_u and X_v. Hence, if X_u has dimension d_u and X_v has dimension d_v, the set B_{uv} contains $d_u \times d_v$ entries. Suppose a tree $\mathcal{G}_{\preceq}$ and its undirected version $\mathcal{G}_{\tilde{\preceq}} = (V, E)$ are given, as well as a multiscale density $q(x|\theta) = N\left(x; 0, Q^\theta\right)$ which factors according to $\mathcal{G}_{\tilde{\preceq}}$. Then, it can be shown that the inverse covariance matrix $\left[Q^\theta\right]^{-1}$ is guaranteed to have zeros in the entries indexed by B_{uv}, if the edge $(u, v) \notin E$. As an example, consider the tree shown in Figure 4.7(a), and assume that $q(x|\theta)$ is a multiscale density which factors according to this tree. Figure 4.7(b) shows the structure of the matrix $\left[Q^\theta\right]^{-1}$, assuming

[14] The notation $N(x; \mu, \Sigma)$ denotes a Gaussian density defined for random vector X which has mean vector μ and covariance matrix Σ.

that each random variable X_v is a scalar. The dark blocks in Figure 4.7(b) represent possible non-zero entries, while the white blocks represent entries which must be zero.

Recall that the goal of optimization problem $\widetilde{\mathcal{Q}}(\Theta)$ is to find the value of $\theta \in \Theta$ such that $D\left(p^*(x_M)\|q(x_M|\theta)\right)$ is minimized. Since $q(x|\theta)$ is a zero-mean Gaussian density, $q(x_M|\theta)$ is a zero-mean Gaussian with covariance Q^θ_M, *i.e.* Q^θ_M contains the entries of Q^θ indexed by the set M. Consequently, the goal of problem $\widetilde{\mathcal{Q}}(\Theta)$ (as considered here) is to find a covariance matrix Q^θ, whose inverse has the zero/non-zero structure indicated by the tree $\mathcal{G}^\sim_{\preceq}$, such that the sub-matrix Q^θ_M "best" matches the target covariance P^*_M. The definition of "best" is of course determined by the cost function $D\left(p^*(x_M)\|q(x_M|\theta)\right)$, which for the zero-mean Gaussian realization problem may be written as follows,

$$D\left(p^*(x_M)\|q(x_M|\theta)\right) = -\frac{d_M}{2} - \frac{1}{2}\log\left(\det\left(\left[Q^\theta_M\right]^{-1} P^*_M\right)\right) + \frac{1}{2}\operatorname{trace}\left(\left[Q^\theta_M\right]^{-1} P^*_M\right), \quad (4.42)$$

with d_M denoting the dimension of random vector X_M. Therefore, the Gaussian realization problem considered here is really a structured matrix optimization problem.

To develop an algorithm for solving this problem, we use the theoretical framework established in the preceding sections of this chapter. Consider now the parameterized alternating minimization procedure discussed in Section 4.3.2, and in particular, consider the set of densities $\{p(x|\gamma)\}_{\gamma\in\Gamma}$ which permit such an alternating procedure. For our purposes, each $p(x|\gamma)$ is a zero-mean Gaussian density with positive-definite covariance matrix P^γ such that the marginal constraint $p(x_M|\gamma) = p^*(x_M)$ is satisfied. Therefore, each $p(x|\gamma)$ has the correct marginal $p^*(x_M)$ but does not necessarily have special factorization structure.

The algorithm developed here is an implementation of the alternating minimizations in (4.25) for the two sets of Gaussian densities $\{p(x|\gamma)\}_{\gamma\in\Gamma}$ and $\{q(x|\theta)\}_{\theta\in\Theta}$ defined in this section. These two sets of densities are particularly well-suited for this iterative approach because of the fact that the mappings $\mathcal{M}(\cdot)$ and $\mathcal{T}(\cdot)$ exist and are simple to characterize, and consequently, the alternating minimizations in (4.25) may equivalently be written in terms of the mappings $\mathcal{M}(\cdot)$ and $\mathcal{T}(\cdot)$ as in (4.26). To understand these two mappings, notice that both collections $\{p(x|\gamma)\}_{\gamma\in\Gamma}$ and $\{q(x|\theta)\}_{\theta\in\Theta}$ only contain Gaussian densities with positive-definite covariance matrices. As a result, the KL divergence $D\left(p(x|\gamma)\|q(x|\theta)\right)$ is finite for all finite choices of $\gamma \in \Gamma$ and $\theta \in \Theta$, a fact which can be seen by applying (4.42). Furthermore, $D\left(p(x|\gamma)\|q(x|\theta)\right)$ is continuous with respect to γ and θ and is equal to zero if and only if $P^\gamma = Q^\theta$ [17].

The preceding facts imply that both optimization problems in (4.25) have unique solutions. Specifically, the iterations in (4.26), defined by the mappings $\mathcal{T}(\cdot)$ and $\mathcal{M}(\cdot)$, may be written as follows

$$\gamma^{(i)}: \quad p(x|\gamma^{(i)}) = \mathcal{F}^M\left(q(x|\theta^{(i-1)})\right), \quad (4.43a)$$

$$\theta^{(i)}: \quad q(x|\theta^{(i)}) = p^T(x|\gamma^{(i)}). \quad (4.43b)$$

Since $\mathcal{M}(\cdot)$ and $\mathcal{T}(\cdot)$ exist in this case, the convergence result stated in Proposition 4.3 indicates that the sequence in (4.43) converges to a fixed point as long as the iterates $\theta^{(i)}$ remain bounded. In addition, Proposition 4.4 indicates that all fixed points are either local extrema or saddle points.

For the Gaussian realization problem considered here, it is difficult to determine if a given initial starting point $\theta^{(0)}$ generates a bounded sequence $\{\theta^{(i)}\}$. This is due to the fact that the set

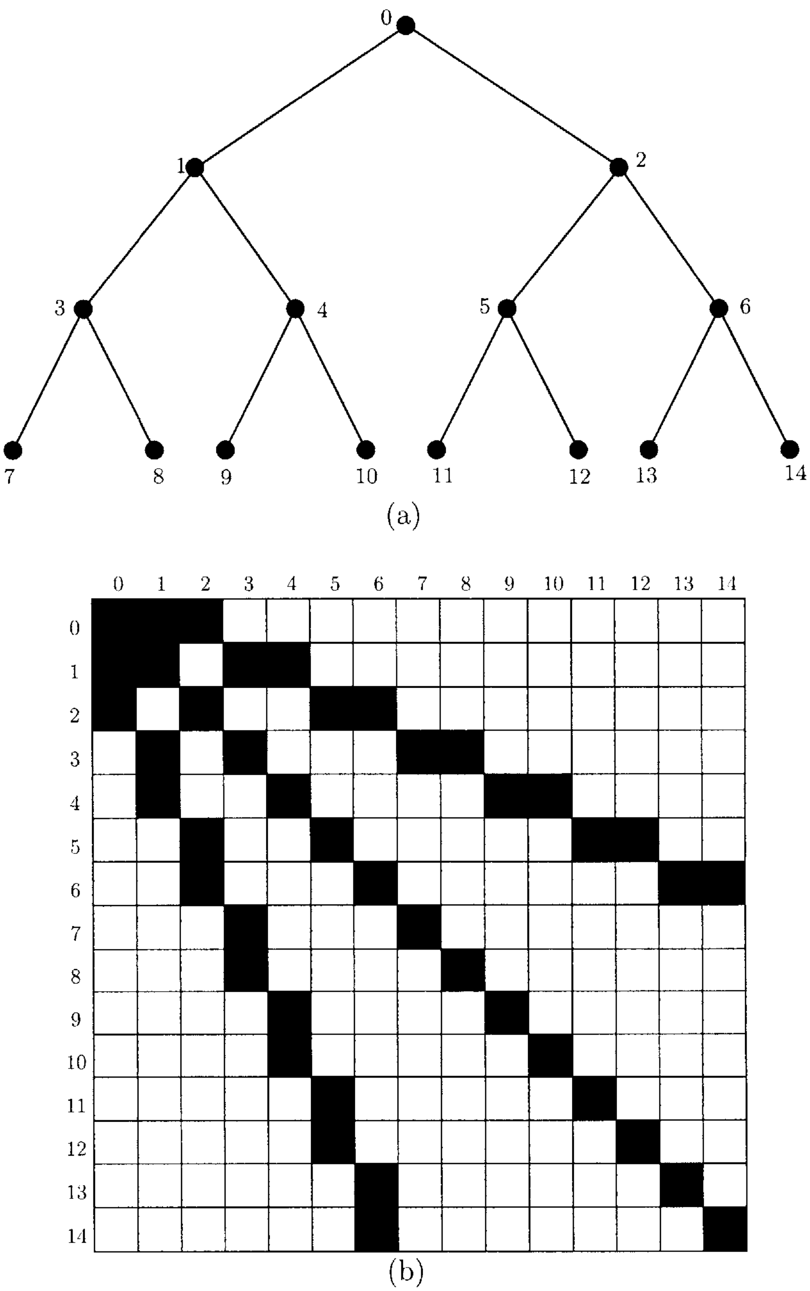


Figure 4.7. (a) An undirected tree $\mathcal{G}^{\sim}_{\preceq} = (V, E)$ with 15 vertices. (b) Given a Gaussian multiscale density $q(x|\theta) = N\left(x; 0, Q^{\theta}\right)$ which factors according to the tree in (a) and where each X_v, $v \in V$, is a scalar random variable, the structure of $\left[Q^{\theta}\right]^{-1}$ is as shown. The dark blocks represent possible non-zero entries, while the white blocks represent entries of $\left[Q^{\theta}\right]^{-1}$ which must be zero.

Θ_0 defined in Proposition 4.3 is unbounded for all initial starting points $\theta^{(0)}$.[15] Therefore, it is possible to have a sequence $\{\theta^{(i)}\}$ for which the cost function $D\left(p^*(x_M)\|q(x_M|\theta^{(i)})\right)$ decreases at each iteration but at the same time $\theta^{(i)} \longrightarrow \infty$. For the moment, we do not consider the issue of boundedness, choosing to focus instead on the implementation details of (4.43). In Section 4.4.3, we provide a rescaled version of the iterations in (4.43) which overcomes this problem and ensures that the iterations always converge to a fixed point.

Notice that the iterations in (4.43) are identical to those derived in (4.3) for the space of all densities. In essence, the algorithm proposed here is an implementation of the iterative approach first described in Section 4.1 but limited to the class of Gaussian densities. This is due to the fact that Gaussian densities map to Gaussian densities under both $p \longrightarrow p^T$ and $q \longrightarrow \mathcal{F}^M(q)$ (assuming $p^*(x_M)$ is also Gaussian). Consequently, given an initial Gaussian multiscale density $q^{(0)}$ with positive-definite covariance, the iterations in (4.3) generate densities which lie in the sets $\{p(x|\gamma)\}_{\gamma\in\Gamma}$ and $\{q(x|\theta)\}_{\theta\in\Theta}$ defined in this section. The iterations in (4.43) then represent a parameterized implementation of (4.3) for the Gaussian multiscale realization problem. Because of this fact, Algorithms 4.1 and 4.2 can be used to efficiently implement (4.43), as the following section demonstrates.

■ 4.4.2 An Efficient Realization Algorithm for Gaussian Multiscale Models

In order to develop an efficient algorithm for solving problem $\widetilde{\mathcal{Q}}(\Theta)$, it is important to parameterize each Gaussian multiscale density $q(x|\theta)$ directly in terms of its edge marginals. To do this, let $\mathcal{G}_{\preceq}$ be a given rooted tree with undirected version $\mathcal{G}_{\preceq}^{\sim} = (V, E)$. Since $q(x|\theta)$ factors in terms of the marginals $q(x_s, x_t|\theta)$, $(s,t) \in E$, we choose to parameterize the set $\{q(x|\theta)\}_{\theta\in\Theta}$ using these local marginals.[16] Specifically, for each edge $(s,t) \in E$, $q(x_s, x_t|\theta)$ is constrained to be a zero-mean Gaussian with a positive-definite covariance matrix, where [17]

$$q(x_s, x_t|\theta) = N\left(x_s, x_t; 0, \begin{bmatrix} Q_s & Q_{s,t} \\ Q_{t,s} & Q_t \end{bmatrix}\right).$$

By parameterizing the densities $q(x|\theta)$ in this manner, we must be mindful of the fact that two marginals $q(x_s, x_t|\theta)$ and $q(x_s, x_u|\theta)$, which share the same variable x_s, must have the same marginal $q(x_s|\theta)$. This marginal requirement simply constrains the sub-matrices Q_s to match in the two marginals $q(x_s, x_t|\theta)$ and $q(x_s, x_u|\theta)$. However, this requirement is not a concern here since the proposed algorithm generates marginals which automatically satisfy this constraint.

Using the preceding parametrization of $q(x|\theta)$, Algorithm 4.2 (see Section 4.2.2) may be used to recursively calculate the parameters of any conditional density $q(x_s|x_t, x_{L_s}, \theta)$ needed to implement the iterations in (4.43), and then, Algorithm 4.1 may be used to incorporate the target density $p^*(x_M)$. The remaining discussion focuses on the matrix equations which result from the recursive calculations involved in Algorithms 4.1 and 4.2. The interested reader should refer to Section C.3 for a more detailed derivation.

[15] See Section 4.4.3 for a more detailed discussion of this issue.

[16] Other parameterizations are possible such as parameterizing the conditional densities $q(x_s|x_t, \theta)$ as previously discussed in Section 2.3.2.

[17] In keeping with the previous notation, the entries of this covariance matrix should really be denoted by Q_s^θ, $Q_{s,t}^\theta$, etc., but we drop the superscript θ for notational simplicity with the understanding that these entries are parameters of the model and indexed by the set Θ.

Algorithm 4.2 Applied to the Gaussian Realization Problem

Examining Algorithm 4.2, we see that the calculations involve four different conditional densities: $q(x_t|x_s)$, $q(x_t|x_{L_t})$, $q(x_t|x_{L_s})$, and $q(x_s|x_t, x_{L_s})$, where vertex t is the parent of vertex s. For the Gaussian problem, we parameterize each of these densities as follows,

$$q(x_t|x_s,\theta) = N\left(x_t; F_s x_s, \tilde{Q}_s\right), \tag{4.44a}$$

$$q(x_t|x_{L_t},\theta) = N\left(x_t; M_t^m x_{L_t}, R_t^m\right), \tag{4.44b}$$

$$q(x_t|x_{L_s},\theta) = N\left(x_t; M_s^p x_{L_s}, R_s^p\right), \tag{4.44c}$$

$$q(x_s|x_t, x_{L_s},\theta) = N\left(x_s; M_s \left(\begin{array}{c} x_t \\ x_{L_s} \end{array} \right), R_s\right). \tag{4.44d}$$

The parameters of the first density may be determined directly from the model parameters associated with the marginal $q(x_s, x_t|\theta)$,

$$F_s = Q_{t,s} Q_s^{-1}, \tag{4.45a}$$

$$\tilde{Q}_s = Q_t - Q_{t,s} Q_s^{-1} Q_{s,t}. \tag{4.45b}$$

The parameters of the second density are calculated by the merge step of Algorithm 4.2 and are given by,

$$R_t^m = \left[(1-q_t) Q_t^{-1} + \sum_{s\in\chi(t)} [R_s^p]^{-1} \right]^{-1}, \tag{4.46a}$$

$$M_t^m = R_t^m \left[\; [R_{s_1}^p]^{-1} M_{s_1}^p \; \middle| \; [R_{s_2}^p]^{-1} M_{s_2}^p \; \middle| \; \cdots \; \middle| \; [R_{s_{q_t}}^p]^{-1} M_{s_{q_t}}^p \; \right], \tag{4.46b}$$

where q_t denotes the number of children of vertex t and where the matrix in (4.46b) is a column-wise concatenation of the matrices $[R_{s_i}^p]^{-1} M_{s_i}^p$ for all child vertices s_i of t.[18] The parameters of the third density are calculated by the prediction step of Algorithm 4.2 and are given by,

$$M_s^p = F_s M_s^m, \tag{4.47a}$$

$$R_s^p = F_s R_s^m F_s^T + \tilde{Q}_s. \tag{4.47b}$$

Finally, the parameters of the fourth density are calculated in the downward pass of Algorithm 4.2 and are given by,

$$M_s = \left[\; J_s \; \middle| \; (I - J_s F_s) M_s^m \; \right], \tag{4.48a}$$

$$R_s = (I - J_s F_s) R_s^m, \tag{4.48b}$$

where $J_s \triangleq R_s^m F_s^T [R_s^p]^{-1}$.

It is important to note that the preceding matrix equations are similar to those derived in [13] for tree-structured recursive estimation. The difference between the recursions here and those in [13] lies in the fact that we must calculate the parameters of the conditional density $q(x_s|x_t, x_{L_s}, \theta)$ in order to then incorporate the target density $p^*(x_M)$. As a result, it is necessary to store and manipulate larger matrices than those considered in [13]. In addition, the covariance matrix R_s is slightly different in our implementation because of the fact that we calculate $q(x_s|x_t, x_{L_s}, \theta)$, as opposed to the implementation in [13] which calculates the covariance of the density $q(x_s|x_M, \theta)$.

[18] The column-wise concatenation of matrices $A_1, A_2, \ldots, A_n$ (each with the same number of rows) generates a $1 \times n$ block matrix, where the first sub-block is A_1, the second sub-block is A_2, and so forth.

Algorithm 4.1 Applied to the Gaussian Realization Problem

Given the parameters M_s and R_s of the conditional density $q(x_s|x_t, x_{L_s}, \theta)$, it is then straightforward to apply Algorithm 4.1 to determine the marginals $p(x_s, x_t|\gamma)$ of the density $p(x|\gamma) = q(x_{V-M}|x_M, \theta)p^*(x_M)$. Algorithm 4.1 essentially consists of two steps: marginalization of densities and multiplication of densities. The marginalization step is simple to implement since it involves picking the appropriate sub-block of a covariance matrix, while the multiplication step can be implemented by matrix multiplication.

As an example, the second step in Algorithm 4.1 requires calculating the marginal $p(x_t, x_{L_s}|\gamma)$ of $p(x_t, x_{L_t}|\gamma)$. Letting $P_{A,B}$ denote the covariance between vectors X_A and X_B under density $p(x|\gamma)$, the marginal $p(x_t, x_{L_t}|\gamma)$ is parameterized as follows,

$$p(x_t, x_{L_t}|\gamma) = N\left(x_t, x_{L_t}; 0, N_t\right), \quad \text{where } N_t \triangleq \begin{bmatrix} P_t & P_{t,L_t} \\ P_{L_t,t} & P_{L_t} \end{bmatrix}. \tag{4.49}$$

The marginal $p(x_t, x_{L_s}|\gamma)$ is then parameterized by

$$p(x_t, x_{L_s}|\gamma) = N\left(x_t, x_{L_s}; 0, N_t^s\right), \quad \text{where } N_t^s \triangleq \begin{bmatrix} P_t & P_{t,L_s} \\ P_{L_s,t} & P_{L_s} \end{bmatrix}, \tag{4.50}$$

and the covariance matrix N_t^s is the sub-block of N_t which corresponds to the random vectors X_t and X_{L_s}.

Consider now the third step of Algorithm 4.1, which requires computing $p(x_s, x_t, x_{L_s}|\gamma) = q(x_s|x_t, x_{L_s}, \theta)p(x_t, x_{L_s}|\gamma)$. Using the parameters M_s and R_s of the density $q(x_s|x_t, x_{L_s}, \theta)$, as well as the covariance matrix N_t^s of the density $p(x_t, x_{L_s}|\gamma)$, the covariance of $p(x_s, x_t, x_{L_s}|\gamma)$ is straightforward to calculate. Specifically, the density $p(x_s, x_t, x_{L_s}|\gamma)$ is parameterized as follows,

$$p(x_s, x_t, x_{L_s}|\gamma) = N\left(x_s, x_t, x_{L_s}; 0, \begin{bmatrix} M_s N_t^s M_s^T + R_s & M_s N_t^s \\ N_t^s M_s^T & N_t^s \end{bmatrix}\right). \tag{4.51}$$

Using the covariance matrix in (4.51), the covariance of the desired marginal $p(x_s, x_t|\gamma)$ may then be determined by choosing the correct sub-block.

Combining Algorithms 4.1 and 4.2

By performing the recursions in Algorithms 4.1 and 4.2 in concert with one another, we now have an efficient means of implementing the iterations in (4.43) for the Gaussian realization problem. The sequence of steps required to complete each iteration of (4.43) is summarized in the following algorithm.

Algorithm 4.3 (Algorithms 4.1 and 4.2 for the Gaussian Realization Problem).
Let $\mathcal{G}_{\preceq}$ be a given rooted tree with undirected version $\mathcal{G}_{\preceq}^{\sim} = (V, E)$. Let M be the set of all leaf vertices of $\mathcal{G}_{\preceq}$, and suppose a zero-mean Gaussian target density $p^*(x_M)$ is specified. Assume the parameters Q_s and $Q_{s,t}$ of the multiscale density $q(x|\theta)$ are specified for all $(s, t) \in E$.

Initialization: For each leaf vertex s and its unique parent t, $q(x_t|x_{L_s}, \theta) \triangleq q(x_t|x_s, \theta)$, and the parameters F_s and $\tilde{Q}_s$ of $q(x_t|x_s, \theta)$ are calculated using (4.45).

Upward Pass: Choose any bottom-up ordering $(u_1, \ldots, u_m)$ on the non-leaf vertices of $\mathcal{G}_{\preceq}$.
FOR $i = 1, \ldots, m$ DO:

(1) Set $t = u_i$.

(2) Compute the parameters M_t^m and R_t^m of $q(x_t|x_{L_t}, \theta)$ using (4.46).

(3) Compute the parameters M_t^p and R_t^p of $q(x_{\pi(t)}|x_{L_t}, \theta)$ using (4.47).

Downward Pass: Choose any top-down ordering $(v_1, \ldots, v_m)$ on the non-leaf vertices of $\mathcal{G}_{\preceq}$. Set $t = v_1$, and compute $p(x_t, x_M|\gamma) = q(x_t|x_M, \theta)p^*(x_M)$ as follows,

$$p(x_t, x_M|\gamma) = N\left(x_t, x_M; 0, \begin{bmatrix} M_t^m P_M^* (M_t^m)^T + R_t^m & M_t^m P_M^* \\ P_M^* (M_t^m)^T & P_M^* \end{bmatrix}\right).$$

FOR $i = 2, \ldots, m$ DO:

(1) Set $s = v_i$, and set t equal to the unique parent of s.

(2) Compute the parameters M_s and R_s of $q(x_s|x_t, x_{L_s}, \theta)$ using (4.48).

(3) Marginalize $p(x_t, x_{L_t}|\gamma)$ to get $p(x_t, x_{L_s}|\gamma)$, *i.e.* choose the sub-block N_t^s of the covariance N_t as shown in (4.49) and (4.50).

(4) Compute $p(x_s, x_t, x_{L_s}|\gamma) = q(x_s|x_t, x_{L_s}, \theta)p(x_t, x_{L_s}|\gamma)$ using (4.51).

(5) Marginalize $p(x_s, x_t, x_{L_s}|\gamma)$ to get $p(x_s, x_t|\gamma)$ and $p(x_s, x_{L_s}|\gamma)$. ◄

Consider now the asymptotic computational complexity of Algorithm 4.3. We assume that the dimension d_v of all random vectors X_v, $v \in V$, is chosen independent of the problem size N, where N denotes the dimension of the target random vector X_M. Because of this assumption, all matrix inversions computed in Algorithm 4.3 are asymptotically negligible, since all such inversions involve matrices of size $d_v \times d_v$, *i.e.* the size of the state vector. The most significant component of the computational complexity comes from matrix multiplies, such as those performed in (4.51). For example, if $t = v_0$ is the root vertex, calculating the density $p(x_t, x_M|\gamma)$ requires multiplying the $d_t \times N$ matrix $M_t = M_t^m$ by the $N \times N$ matrix P_M^*, which requires approximately $d_t N^2$ multiplies. Asymptotically, this is the dominant calculation, and consequently, Algorithm 4.3 has computational complexity $\mathcal{O}(N^2)$ (per iteration) for a problem of size N.

4.4.3 A Rescaling Algorithm for the Gaussian Multiscale Realization Problem

As previously mentioned in Section 4.4.1, the iterations in (4.43) are not guaranteed to converge to a fixed point in the case of the Gaussian realization problem. Specifically, for every initial starting point $\theta^{(0)}$, the set Θ_0 defined in Proposition 4.3 is unbounded, and consequently, there is no guarantee on the boundedness of any particular sequence $\{\theta^{(i)}\}$. To address this issue, this section proposes a "rescaled" version of the iterations in (4.43), which is guaranteed to generate a bounded sequence as long as $\theta^{(0)}$ is chosen so that the initial cost $\varepsilon_0 < \infty$. The following two examples illustrates these ideas.

Example 4.3 (Unbounded Sequences in the Gaussian Realization Problem).
Consider the multiscale model shown previously in Figure 4.5. This example demonstrates the fact that Θ_0 is an unbounded set for every starting point $\theta^{(0)}$ of the iterations in (4.43). To do this,

consider two different starting points $\theta^{(0)}$ and $\bar{\theta}^{(0)}$, where the corresponding multiscale densities $q(x|\theta^{(0)})$ and $q(x|\bar{\theta}^{(0)})$ are both zero-mean and Gaussian. For notational convenience, we assume that $q(x|\theta^{(0)})$ and $q(x|\bar{\theta}^{(0)})$ have covariances Q and $\bar{Q}$ respectively and that the marginal densities $q(x_M|\theta^{(0)})$ and $q(x_M|\bar{\theta}^{(0)})$ have marginal covariances Q_M and $\bar{Q}_M$ respectively.

Consider first the marginal covariance Q_M, which can be shown to be the following function of the state covariances Q_v and edge covariances Q_{uv} of $q(x|\theta^{(0)})$,

$$Q_M = \begin{bmatrix} Q_1 & Q_{10}Q_0^{-1}Q_{02} \\ Q_{20}Q_0^{-1}Q_{01} & Q_2 \end{bmatrix}. \tag{4.52}$$

Now, suppose the covariances $\bar{Q}_v$ and $\bar{Q}_{uv}$ of $q(x|\bar{\theta}^{(0)})$ are defined in terms of the covariances Q_v and Q_{uv} as follows,

$$\bar{Q}_1 \triangleq Q_1, \quad \bar{Q}_2 \triangleq Q_2, \quad \bar{Q}_0 \triangleq \alpha Q_0, \tag{4.53a}$$

$$\bar{Q}_{10} \triangleq Q_{10}, \quad \bar{Q}_{20} \triangleq \alpha Q_{20}, \tag{4.53b}$$

for some $\alpha \in (0, \infty)$. Using (4.52), it can be seen that $\bar{Q}_M = Q_M$ for all choices of $\alpha \in (0, \infty)$. Therefore, even though the two densities $q(x|\theta^{(0)})$ and $q(x|\bar{\theta}^{(0)})$ are different, the marginals $q(x_M|\theta^{(0)})$ and $q(x_M|\bar{\theta}^{(0)})$ are identical due to the form of the bijection $\theta^{(0)} \longleftrightarrow \bar{\theta}^{(0)}$ in (4.53).

One immediate consequence of this fact is that it proves that the set Θ_0 is unbounded. Since $q(x_M|\theta^{(0)}) = q(x_M|\bar{\theta}^{(0)})$, the KL divergences $D\left(p^*(x_M)\|q(x_M|\theta^{(0)})\right)$ and $D\left(p^*(x_M)\|q(x_M|\bar{\theta}^{(0)})\right)$ are equal for all values of $\alpha \in (0, \infty)$. For each positive integer N, consider the collection of all such $\bar{\theta}^{(0)}$ for which $\alpha \le N$, *i.e.* $\Phi_N \triangleq \{\bar{\theta}^{(0)}|\alpha \in (0, N]\}$. Taking the limit $N \longrightarrow \infty$ gives the unbounded set Φ_∞. The set $\Theta_0 = \left\{\theta \in \Theta \,|D\left(p^*(x_M)\|q(x_M|\theta)\right) \le D\left(p^*(x_M)\|q(x_M|\theta^{(0)})\right) \right\}$ is a superset of Φ_∞ and is therefore unbounded as well. ◀

The ideas presented in the preceding example may be generalized to the class of all Gaussian multiscale models. Specifically, it is important to note that every multiscale density $q(x|\theta)$ has an infinite number of corresponding multiscale densities $q(x|\theta')$, each possessing the same marginal $q(x_M|\theta) = q(x_M|\theta')$. Because of this redundancy, the iterations in (4.43) may become poorly scaled in the limit.

To deal with this issue, we would ideally like to identify an equivalence class $\langle\theta\rangle$ with each multiscale model $q(x|\theta)$, such that $\theta' \in \langle\theta\rangle$ if and only if $q(x_M|\theta) = q(x_M|\theta')$. Given such a partitioning of the set Θ, the issue of scaling can be overcome by performing the iterations in (4.43) with respect to the set of equivalence classes. Unfortunately, characterizing these equivalence classes can be difficult for the multiscale realization problem, and as such, we propose a partitioning which deals with the issue of scaling but does not necessarily correspond to an equivalence relation. The following example demonstrates this idea.

Example 4.4 (Rescaling the Covariances of Multiscale Models).
Consider again the multiscale model and the densities $q(x|\theta^{(0)})$ and $q(x|\bar{\theta}^{(0)})$ discussed in Example 4.3. In this example, we propose a more general rescaling of the covariance $\bar{Q}$ than that provided in (4.53). In particular, consider the following bijection between $\theta^{(0)}$ and $\bar{\theta}^{(0)}$ where M is an arbitrary invertible matrix,

$$\bar{Q}_1 \triangleq Q_1, \quad \bar{Q}_2 \triangleq Q_2, \quad \bar{Q}_0 \triangleq M^T Q_0 M, \tag{4.54a}$$

$$\bar{Q}_{10} \triangleq Q_{10}M, \quad \bar{Q}_{20} \triangleq Q_{20}M. \tag{4.54b}$$

Using (4.52), it can be seen that any such choice for M generates a covariance $\bar{Q}$ which satisfies $\bar{Q}_M = Q_M$. Therefore, (4.54) provides a rather general method for rescaling the covariances of the multiscale model in Figure 4.5, without changing the marginal density $q(x_M|\theta^{(0)})$.

In this section, we focus on a particular choice for the matrix M in (4.54); specifically, we set $M = Q_0^{-1/2}$, where $Q_0^{-1/2}$ is the inverse symmetric square-root of Q_0. This choice for M gives the following marginal covariances of $\bar{Q}$,

$$\bar{Q}_1 \triangleq Q_1, \quad \bar{Q}_2 \triangleq Q_2, \quad \bar{Q}_0 \triangleq I, \tag{4.55a}$$

$$\bar{Q}_{10} \triangleq Q_{10}Q_0^{-1/2}, \quad \bar{Q}_{20} \triangleq Q_{20}Q_0^{-1/2}. \tag{4.55b}$$

Therefore, this particular rescaling generates a multiscale model with the root covariance $\bar{Q}_0$ equal to the identity.

The bijection defined in (4.55) may be extended to more complicated multiscale models, as we soon demonstrate. In particular, we will show that any Gaussian multiscale density $q(x|\theta)$ with non-singular covariance Q may be mapped to a multiscale density $q(x|\bar{\theta})$ with a covariance $\bar{Q}$ which satisfies $\bar{Q}_v = I$ for all non-leaf vertices v and in addition satisfies $\bar{Q}_M = Q_M$. Therefore, applying such a mapping to Q generates a multiscale model with the same desired marginal but with each state covariance rescaled to the identity.

Since multiscale models may be rescaled in such a manner, we now focus on the subclass of multiscale models with state covariances equal to the identity. This subclass of multiscale models is sufficiently rich for our purposes because any solution to problem $\widetilde{\mathcal{Q}}(\Theta)$ may be mapped to a corresponding solution which has identity state covariances. In addition, by requiring the state covariances to be equal to the identity, the model parameters are constrained in such a way as to guarantee that the iterations in (4.43) converge, as we soon show. ◀

As Example 4.4 demonstrates, a simple three-vertex Gaussian multiscale model may be rescaled so that the root covariance equals the identity, and this rescaling does not alter the marginal of interest. We now show how the mapping in (4.55) may be generalized to all Gaussian multiscale models defined on arbitrary trees $\mathcal{G}^{\sim}_{\prec} = (V, E)$. To do this, let $q(x|\theta)$ and $q(x|\bar{\theta})$ be two multiscale densities with covariances Q and $\bar{Q}$ respectively. The mapping of interest to us is given by the following,

$$\bar{Q}_v \triangleq \begin{cases} I, & v \in V - M \\ Q_v, & v \in M \end{cases} \tag{4.56a}$$

$$\bar{Q}_{uv} \triangleq \begin{cases} Q_u^{-1/2} Q_{uv} Q_v^{-1/2}, & (u,v) \in E, \ \ u,v \in V - M \\ Q_{uv} Q_v^{-1/2}, & (u,v) \in E, \ \ u \in M \end{cases} \tag{4.56b}$$

where $Q_u^{-1/2}$ and $Q_v^{-1/2}$ are the inverse symmetric square roots of Q_u and Q_v respectively. In (4.56a), all non-leaf state covariances are mapped to the identity, while all leaf covariances remain unchanged, and in (4.56b), the cross-covariances along edges (u, v) are rescaled using the matrices $Q_u^{-1/2}$ and $Q_v^{-1/2}$.

For notational convenience, we denote the mapping in (4.56) by $\mathcal{R}(\cdot)$ since it acts as a rescaling operation on the set of multiscale models. More specifically, $\mathcal{R} : \Theta \longrightarrow \bar{\Theta}$, where $\bar{\Theta}$ denotes the subset of Θ for which each multiscale density $q(x|\theta)$ has state covariances equal to the identity. As the following proposition indicates, the mapping $\mathcal{R}(\cdot)$ satisfies our previously stated criterion in that it scales the state covariances but does not alter the marginal of interest.

Proposition 4.5 (Rescaling Multiscale Models to Have Identity State Covariances).
Let two zero-mean Gaussian multiscale densities $q(x|\theta) = N(x; 0, Q)$ and $q(x|\bar{\theta}) = N(x; 0, \bar{Q})$ be specified such that the marginals $\bar{Q}_v$ and $\bar{Q}_{uv}$ of $\bar{Q}$ are defined in terms of the marginals Q_v and Q_{uv} of Q according to (4.56). Then, $\bar{Q}_v = I$ for all non-leaf vertices v, and the two marginal covariances Q_M and $\bar{Q}_M$ are identical.

Proof. See Appendix C.4. ■

Having defined the rescaling mapping $\mathcal{R}(\cdot)$ in (4.56), we now utilize this mapping in the context of the multiscale realization problem. Simply stated, we use $\mathcal{R}(\cdot)$ to rescale the EM iterations in (4.43). As a result, the problem of unbounded sequences is not an issue since this rescaling is guaranteed to generate a bounded sequence $\{\bar{\theta}^{(i)}\}$. Since $\mathcal{R}(\cdot)$ does not change the marginal of interest, the sequence of KL divergences $\bar{\varepsilon}_i \triangleq D\left(p^*(x_M)\|q(x_M|\bar{\theta}^{(i)})\right)$ is monotonically decreasing, and consequently, the sequence $\{\bar{\theta}^{(i)}\}$ is guaranteed to converge in the limit. We elaborate on these ideas in the remainder of this section.

Consider the following modification of the iterations in (4.43) which has an additional rescaling step,

$$\begin{aligned}
\gamma^{(i)} &: \quad p(x|\gamma^{(i)}) = \mathcal{F}^M\left(q(x|\bar{\theta}^{(i-1)})\right), && \text{(4.57a)}\\
\theta^{(i)} &: \quad q(x|\theta^{(i)}) = p^T(x|\gamma^{(i)}), && \text{(4.57b)}\\
\bar{\theta}^{(i)} &: \quad \bar{\theta}^{(i)} = \mathcal{R}(\theta^{(i)}). && \text{(4.57c)}
\end{aligned}$$

The EM iterations in (4.57a) and (4.57b) have the same form as before, but in addition, the value of $\theta^{(i)}$ is transformed in (4.57c) using $\mathcal{R}(\cdot)$. As a result, the corresponding multiscale density $q(x|\theta^{(i)})$ is mapped to a new density $q(x|\bar{\theta}^{(i)})$ with state covariances equal to the identity. Since this rescaling operation is performed at each iteration, the sequence $\{\bar{\theta}^{(i)}\}$ remains bounded, and in particular, each $\bar{\theta}^{(i)}$ lies in the bounded set[19]

$$\bar{\Theta}_0 \triangleq \left\{\bar{\theta} \in \bar{\Theta} \,\middle|\, D\left(p^*(x_M)\|q(x_M|\bar{\theta})\right) \leq D\left(p^*(x_M)\|q(x_M|\bar{\theta}^{(0)})\right)\right\},$$

as the proof of Proposition 4.6 (to follow) demonstrates. The set $\bar{\Theta}_0$ is analogous to the set Θ_0 introduced in Proposition 4.3, but in this case, we restrict our attention to parameters which lie in the set $\bar{\Theta}$.

Using the fact that $\{\bar{\theta}^{(i)}\}$ is bounded, in addition to the fact that $\{\bar{\varepsilon}_i\}$ is a monotonically decreasing sequence (see Appendix C.5), the result previously provided in Proposition 4.3 indicates that $\{\bar{\theta}^{(i)}\}$ converges to a fixed point $\hat{\theta}$. The following proposition states this fact for the special case of the Gaussian realization problem considered here.

Proposition 4.6 (Convergence of the Sequence $\{\bar{\theta}^{(i)}\}$ for the Gaussian Realization Problem).
Let Θ index the set of zero-mean Gaussian multiscale densities with positive-definite covariances, and suppose $\bar{\theta}^{(0)}$ is an initial starting point for the sequence $\{\bar{\theta}^{(i)}\}$ in (4.57) which satisfies $\bar{\varepsilon}_0 = D\left(p^*(x_M)\|q(x_M|\bar{\theta}^{(0)})\right) < \infty$. Then, the sequence $\{\bar{\theta}^{(i)}\}$ converges to a fixed point $\hat{\theta}$ which satisfies $\hat{\theta} = \mathcal{T}(\mathcal{M}(\hat{\theta}))$.

[19] Recall from the preceding discussion that $\bar{\Theta}$ is defined to be the subset of all Θ where each multiscale density $q(x|\bar{\theta})$, $\bar{\theta} \in \bar{\Theta}$, has non-leaf state covariances equal to the identity.

Proof. See Appendix C.5. ■

The result in Proposition 4.6 is significant because it indicates that the iterations in (4.57) are guaranteed to converge to a fixed point which satisfies the necessary conditions for solutions to the Gaussian multiscale realization problem. Furthermore, it is straightforward to incorporate the rescaling step (4.57c) into the recursive algorithm previously discussed in Section 4.4.2 by simply applying the mapping $\mathcal{R}(\cdot)$ to $\theta^{(i)}$ at each iteration. Since the mapping $\mathcal{R}(\cdot)$ is only a function of edge and vertex marginals, this additional step does not alter the computational complexity of the resulting algorithm. The following section provides several examples which demonstrate the convergence properties of this particular algorithm.

■ 4.4.4 Examples and Results

In this section, we provide several examples which illustrate the convergence properties of the iterative technique developed in this chapter for solving the Gaussian multiscale realization problem. Our goal is to solve the Gaussian realization problem introduced in Section 4.4.1 for several different choices of the target covariance matrix P_M^*. To do this, we use Algorithm 4.3 to recursively compute the necessary marginals at each iteration. In addition, at each iteration we perform the rescaling operation $\mathcal{R}(\cdot)$ discussed in Section 4.4.3 in order to ensure that the model parameters do not diverge to infinity.

A First-Order Markov Process

In this example, we demonstrate the performance of the iterations in (4.57) on a simple first-order Markov process. Consider the 16 point first-order Markov process Y whose conditional independence structure is given by the graphical model shown in Figure 4.8(a), and assume that Y is mapped to vectors X_1 and X_2 as shown in Figure 4.8(b). The goal of this example is to find a scalar variable X_0 such that X_1 and X_2 are conditionally independent given X_0. Since Y is a first-order Markov process, we know that at least two exact solutions, namely $X_0 = Y_8$ and $X_0 = Y_9$, exist for this problem. As we soon discuss, the iterations in (4.57) find solutions which are a linear combination of Y_8 and Y_9.

As a specific example, suppose the entries of P_M^* are specified by the height of the surface in Figure 4.9(a). The magnitudes of the entries of $[P_M^*]^{-1}$ are also provided in Figure 4.9(b). As the latter figure demonstrates, the zero entries (i,j) of $[P_M^*]^{-1}$ precisely correspond to edges (i,j) not present in the graph shown in Figure 4.8(a), and therefore, P_M^* is indeed a first-order Markov process.

Figure 4.10 illustrates the convergence characteristics of (4.57) for this example. Each line in the plot corresponds to the iterates generated by choosing a different initial multiscale parametrization $\bar{\theta}^{(0)}$. Since an exact solution exists in this case, any global minima of optimization problem $\widetilde{Q}(\Theta)$ also corresponds to an exact solution of the realization problem. As Figure 4.10 demonstrates, at least five of the initial starting points produce iterates which converge to exact solutions. At the same time, however, the rate of convergence for each of these starting points is highly variable. In fact, one of the starting points generates a sequence which decreases the cost function only slightly after 1000 iterations. We conjecture that this particular starting point is located in an extremely flat region of the cost function and is also in close proximity to a saddle point. As a result, it requires a large number of iterations to move away from this saddle point.

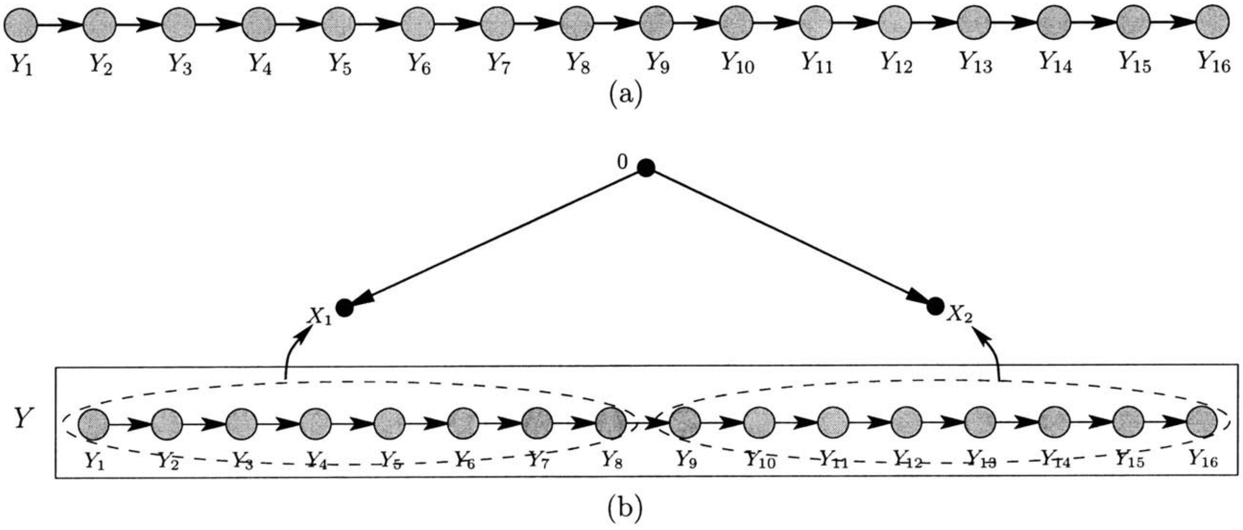

Y_1 Y_2 Y_3 Y_4 Y_5 Y_6 Y_7 Y_8 Y_9 Y_{10} Y_{11} Y_{12} Y_{13} Y_{14} Y_{15} Y_{16}
(a)
0
X_1
X_2
Y
Y_1 Y_2 Y_3 Y_4 Y_5 Y_6 Y_7 Y_8 Y_9 Y_{10} Y_{11} Y_{12} Y_{13} Y_{14} Y_{15} Y_{16}
(b)

Figure 4.8. (a) A 16 point first-order Markov process, Y. (b) Mapping of the process Y to the leaf vertices of a rooted tree with three vertices.

For the starting points in Figure 4.10 which generate sequences converging to an exact solution, we observe that the value of X_0 at a point near convergence is of the form $X_0 = aY_8 + bY_9$, for some choice of a and b. Recall that we also observed this type of solution in Example 4.2 where an infinite "trough" of global minima existed in addition to the obvious trivial solutions. Of course, since we perform a rescaling step at each iteration, the solutions $X_0 = aY_8 + bY_9$ are constrained in this example because the variance of X_0 is required to equal 1.

Fractional Brownian Motion

In this example, we examine the convergence characteristics of the iterations in (4.57) when the target density $p^*(x_M)$ does not have special factorization structure, such as that exhibited by the preceding first-order Markov process. In particular, consider the 256×256 covariance matrix P_M^* whose entries are represented by the surface plot in Figure 4.11(a). Here, P_M^* is the covariance matrix for 256 samples of fractional Brownian motion (fBm) on the interval $(0, 1]$ and with Hurst parameter $H = 0.3$. The log-magnitudes of the entries of $[P_M^*]^{-1}$ are also provided in Figure 4.11(b). The structure of the inverse is interesting in that the magnitudes of the entries decrease very rapidly away from the main diagonal, but none of the entries is equal to zero. Consequently, $p^*(x_M)$ does not have special factorization structure.

In this example, we consider a multiscale model whose underlying tree $\mathcal{G}_{\prec}^{\sim}$ is shown in Figure 4.12. Since the tree has 32 leaf vertices, we set the dimension of X_v to 8 for each leaf vertex v, and we map the 256-dimensional fBm process sequentially to the vectors X_v. For the remaining non-leaf vectors X_v, we constrain each of their dimensions to 4.

Figure 4.13(a) provides convergence curves for 5 different initial starting points. As this plot demonstrates, the curves are almost identical (for very different initial starting points). This observation stands in sharp contrast to the convergence variability seen in the previous example. We conjecture that the degree of variability is a function of the structure of $[P_M^*]^{-1}$. Specifically,

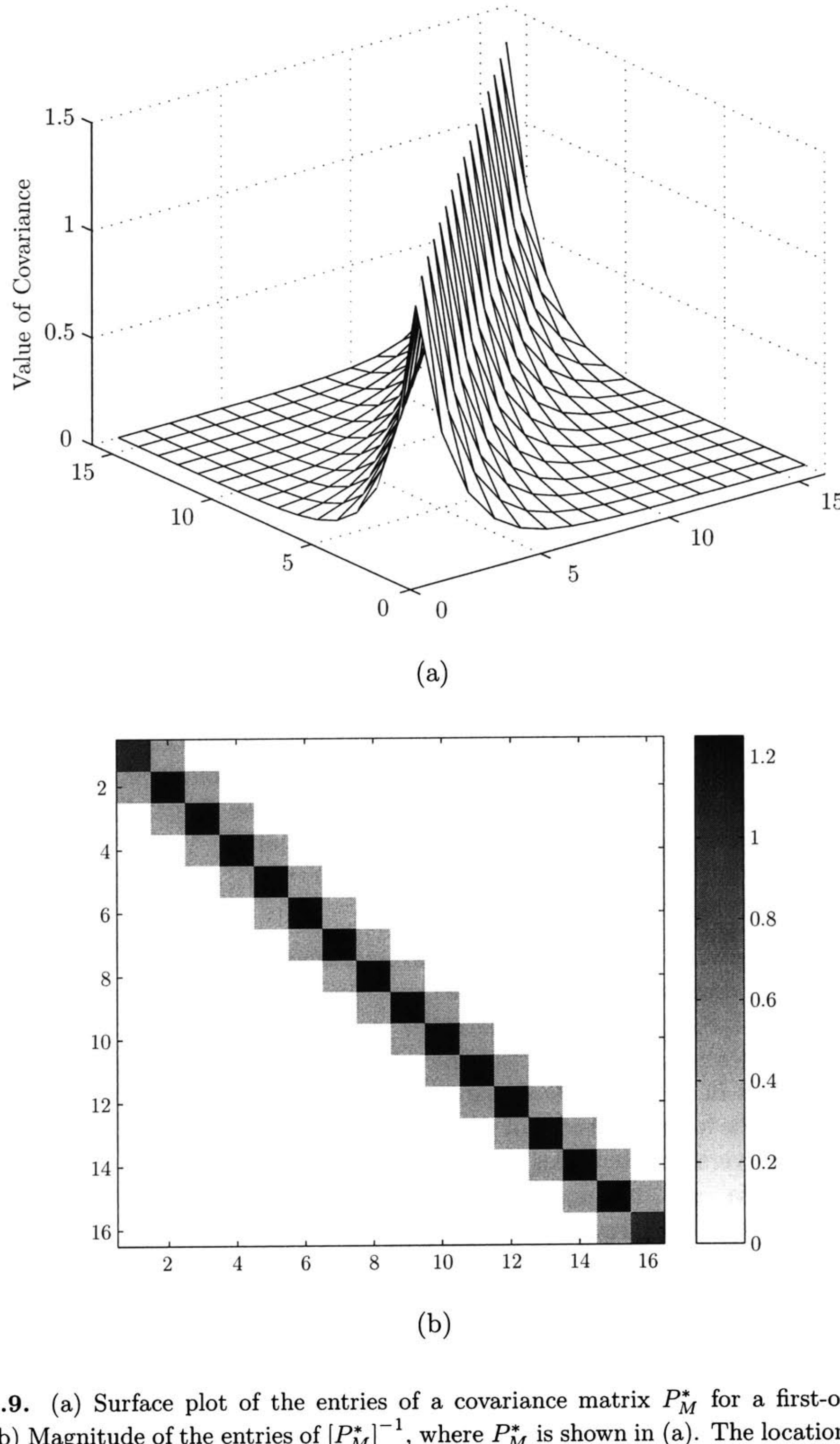


Figure 4.9. (a) Surface plot of the entries of a covariance matrix P_M^* for a first-order Markov process. (b) Magnitude of the entries of $[P_M^*]^{-1}$, where P_M^* is shown in (a). The locations of the zero entries correspond to edges absent in the directed graph shown in Figure 4.8(a).

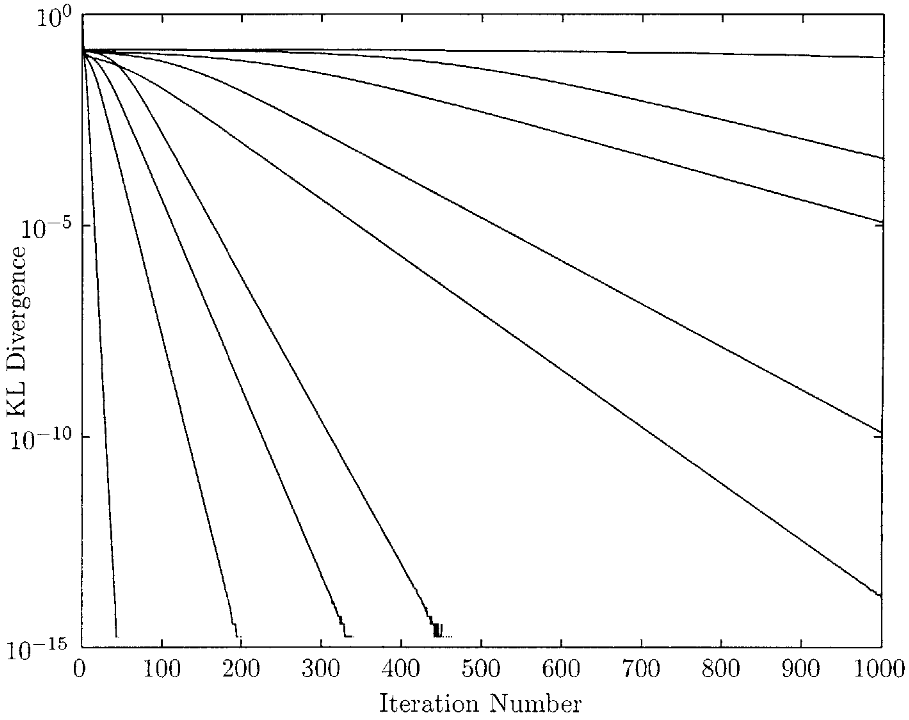
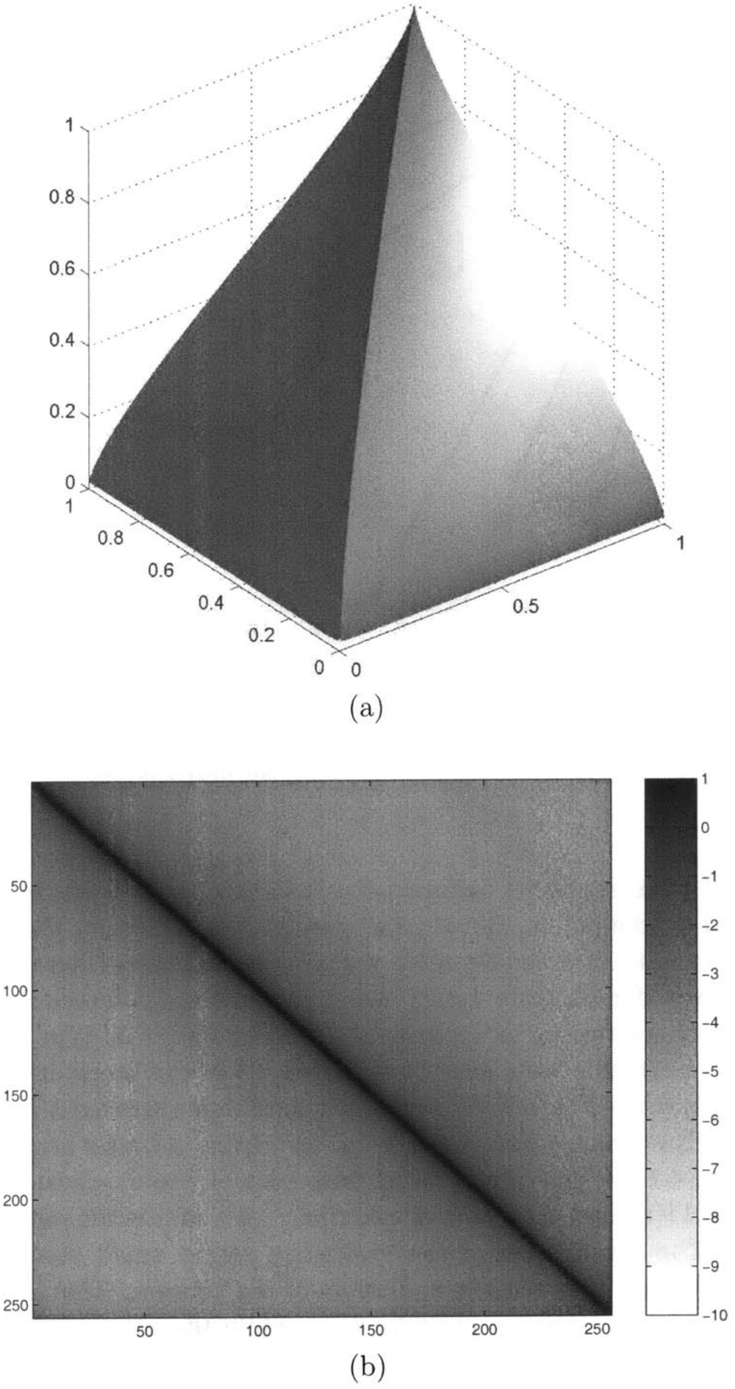


Figure 4.10. Convergence results for the iterations in (4.57) assuming the target covariance P_M^* shown in Figure 4.9(a). Each line corresponds to a different initial starting point.

when $p^*(x_M)$ does not possess special factorization structure, we conjecture that the cost function has a fewer number of saddle points, thereby decreasing the probability of finding a starting point which lingers in the vicinity of a saddle point for a large number of iterations. Notice also that all 5 convergence curves demonstrate rapid initial convergence followed by a much slower rate of convergence. This characteristic is commonly observed with EM-type iterations, and various approaches for dealing with this issue have been addressed in the literature [78].

The convergence curves in Figure 4.13(a) demonstrate that there exist multiscale models which well-approximate the fBm process considered here. To obtain a better sense of the quality of this approximation, Figure 4.13(b) provides a point-wise plot of the absolute error between the true covariance in Figure 4.11(a) and the realized covariance of a multiscale model obtained after 1000 iterations of (4.57). The magnitudes of the errors are rather small overall. The largest errors occur in the cross-covariance between the endpoints of the process. This is most likely due to the fact that one endpoint has a variance close to zero, while the other endpoint has a variance of one. The remaining errors of notable size occur at the major tree boundaries where two points are spatially close in the underlying process but are far apart in the multiscale model. This latter phenomenon has been noted and discussed in a number of sources including [38,49,73], and at least two approaches for dealing with this issue have been proposed [29,103].



(a)

(b)

Figure 4.11. (a) Surface plot of the entries of a 256 × 256 covariance matrix P_M^* which corresponds to fractional Brownian motion with Hurst parameter $H = 0.3$. (b) Log-magnitude of the entries of $[P_M^*]^{-1}$, where P_M^* is shown in (a). Since the entries of the inverse are not zero anywhere, this process factors according to a complete graph, *i.e.* it possesses no conditional independence structure.

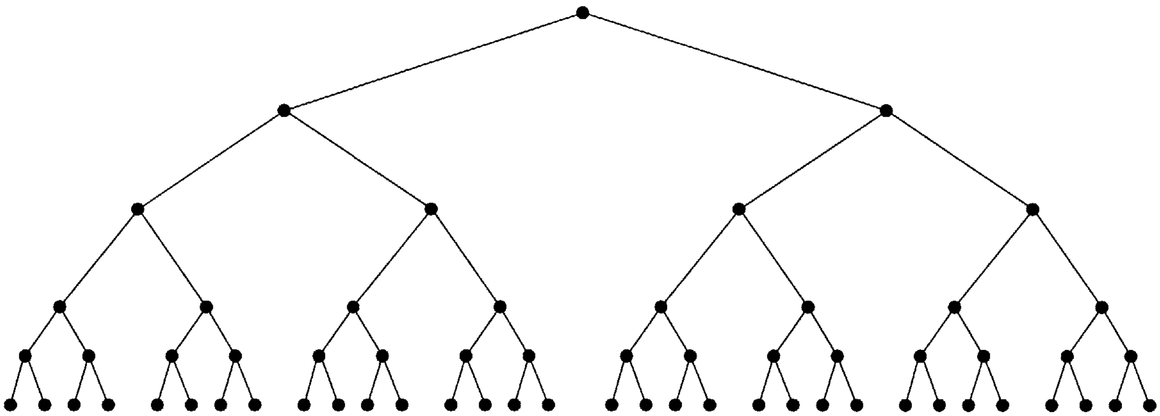


Figure 4.12: Graph structure of the multiscale model to be realized in the fBm example.

A Sinusoidal Covariance

In this final example, we consider the performance of the iterations in (4.57) on a target covariance whose inverse structure lies somewhere between the structures of the first two examples. Consider the 512×512 target covariance matrix P_M^* shown in Figure 4.14(a) and the log-magnitude of $[P_M^*]^{-1}$ shown in Figure 4.14(b). The entries of P_M^* are given by a damped sinusoidal function, where the variances on the main diagonal are equal to 1 and the magnitudes of the cross-covariances decay as a function of the distance from the main diagonal. The structure of the inverse covariance $[P_M^*]^{-1}$ is similar in that the magnitudes of the entries decay rapidly as the distance from the main diagonal increases. In fact, the degree of decay is much more rapid than that seen in the fBm example, and as Figure 4.14(b) illustrates, the inverse is approximately equal to a banded matrix of width k. Because of this, the sinusoidal covariance shown in Figure 4.14(a) is approximately equal to a k^{th}-order Markov process. However, since none of the entries is exactly equal to zero, the target density $p^*(x_M)$ does not possess special factorization structure.

The multiscale model considered in this example is similar to the one previously shown in Figure 4.12, except that one additional layer of vertices is added to give a total of 64 leaf vertices. The 512-dimensional process of interest is sequentially mapped to the random vectors X_v at the 64 leaf vertices such that each X_v has dimension 8. We consider the performance of the iterations in (4.57) when we constrain the state dimensions of all non-leaf vertices to either be 4 or 1.

Figure 4.15(a) shows the convergence results if each non-leaf vector X_v has dimension 4, while Figure 4.15(b) shows the results if each non-leaf vector X_v has dimension 1. Noticeably, the curves are very similar within each plot and between the two plots. As in the previous example, we conjecture that the results are similar within each plot due to the fact that $[P_M^*]^{-1}$ has no zero entries and therefore $p^*(x_M)$ has no special factorization structure.

The fact that the results are similar between the two plots indicates that in this particular example state dimensions of 1 generate as good of an approximation as state dimensions of 4. The choice of state dimension is an important part of the multiscale realization problem which we have not focused on in this thesis. A natural extension of the optimization problem $\tilde{Q}(\Theta)$ is to incorporate the choice of state dimension into the objective function, rather than imposing a fixed hard constraint as we have done here. In this way, the objective of the multiscale realization problem is to find the "best" approximate model with respect to some additional complexity constraints. Some ideas regarding this particular issue have been expressed in [38].

To obtain a better sense of the quality of the approximation obtained in this example, Figure 4.16 shows the magnitudes of the point-wise differences between the true covariance in Figure 4.14(a) and two different approximate solutions. Figure 4.16(a) shows the error for an approximate model generated after 1000 iterations of (4.57), where each non-leaf state dimension is equal to 4. Figure 4.14(b) provides a similar plot but for an approximate model with non-leaf state dimensions equal to 1. The magnitudes of the errors are very similar, with Figure 4.14(b) appearing to have slightly smaller point-wise errors than Figure 4.14(a). This, however, is due to the fact that neither approximation has truly converged to a fixed point.

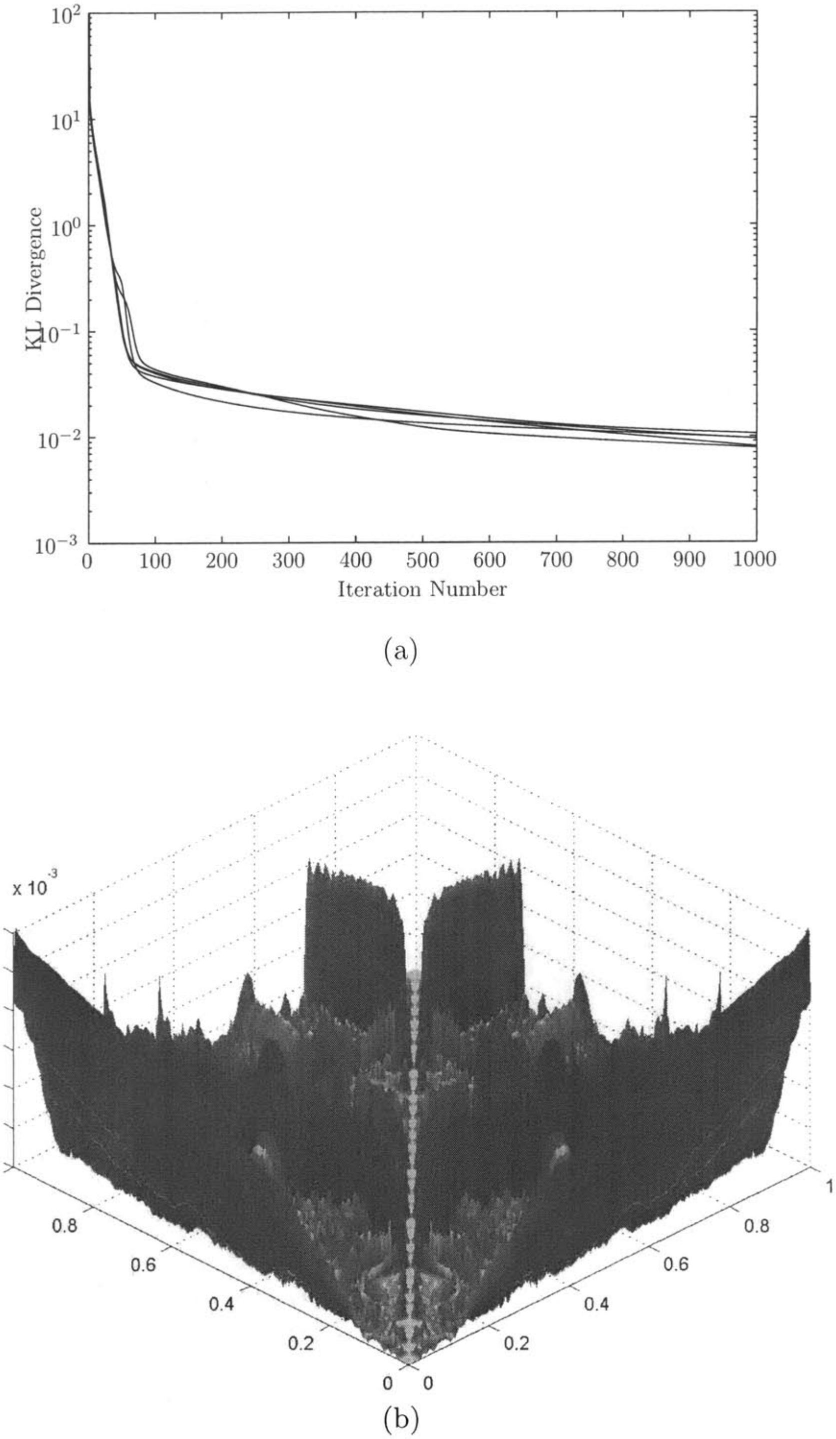
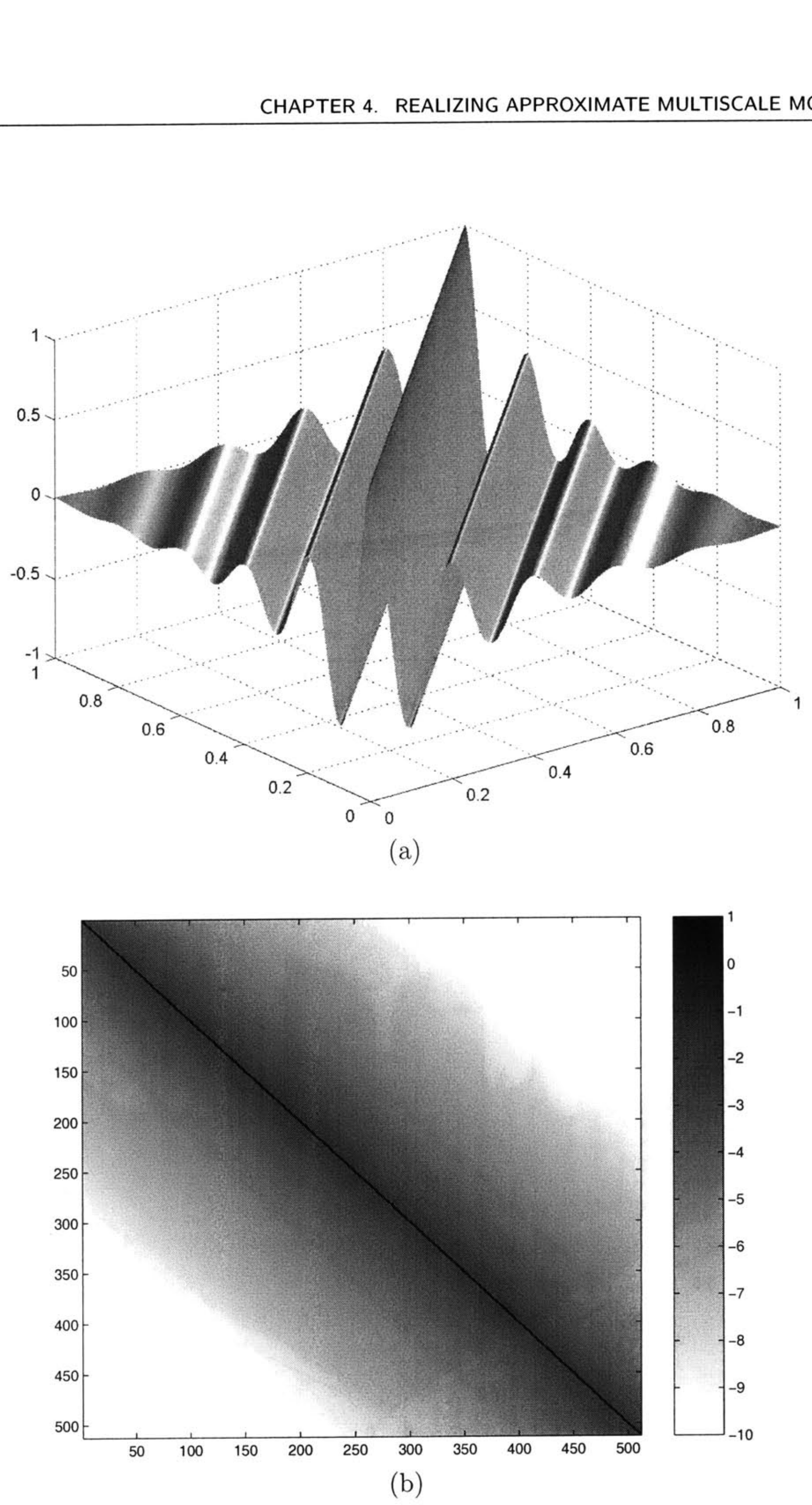


Figure 4.13. (a) Convergence results for the iterations in (4.57) assuming the target covariance P_M^* shown in Figure 4.11(a). Each line corresponds to a different initial starting point. (b) Shows the absolute error between the true covariance in Figure 4.11(a) and an approximate solution generated after 1000 iterations of (4.57). In this example, the dimension of X_v is equal to 4 for each non-leaf vertex v and equal to 8 for each leaf vertex v.



(a)

(b)

Figure 4.14. (a) Surface plot of the entries of a 512×512 covariance matrix P_M^* which corresponds to a damped sinusoid. (b) Log-magnitude of the entries of $[P_M^*]^{-1}$, where P_M^* is shown in (a).

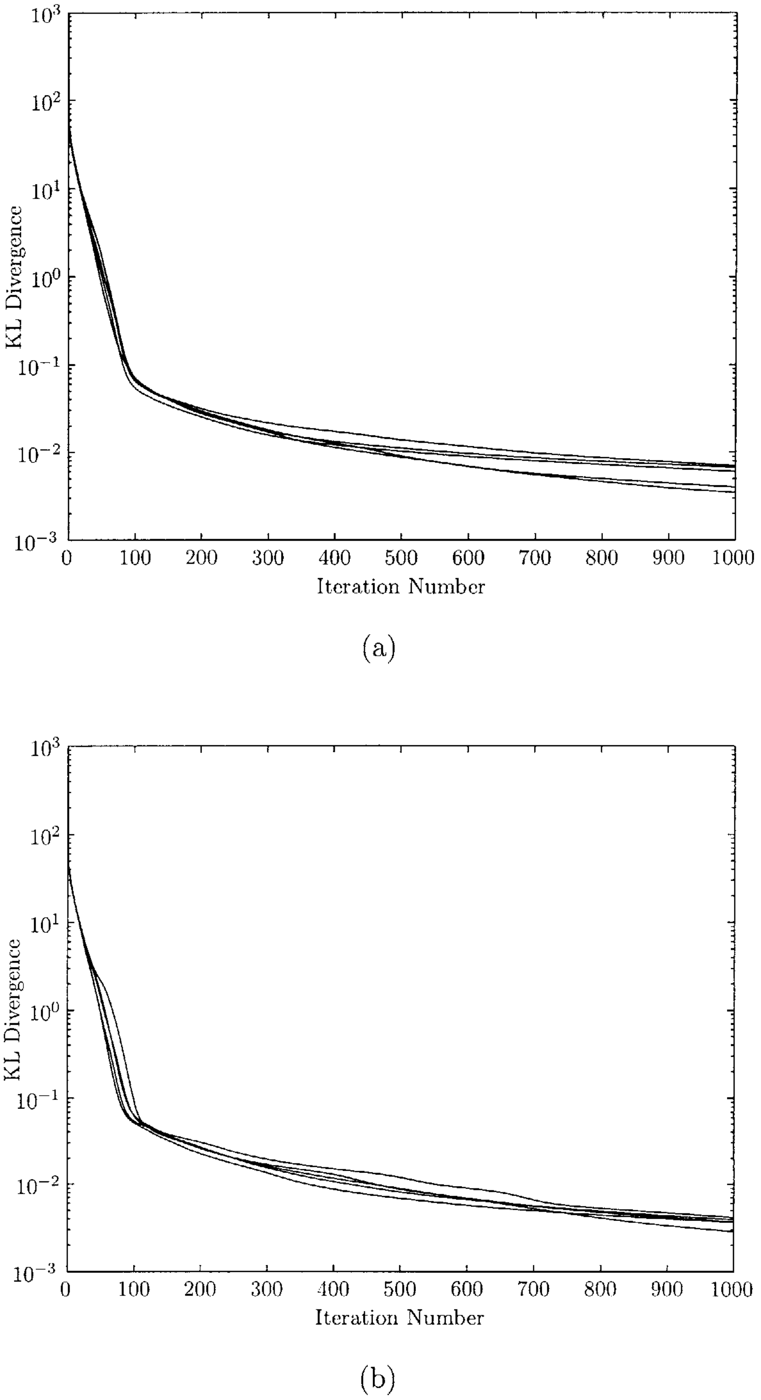
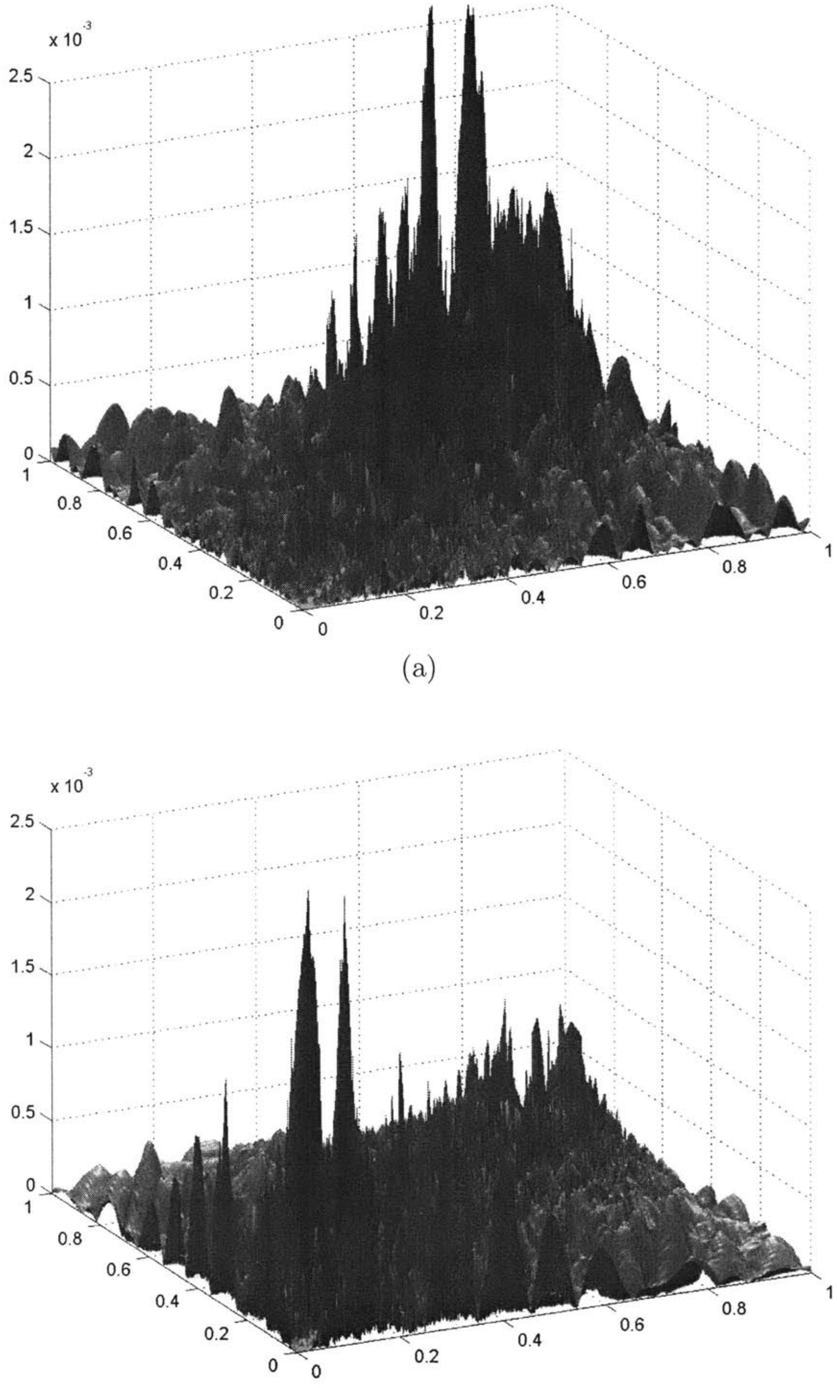


Figure 4.15. (a) Convergence results for the iterations in (4.57) assuming the target covariance P_M^* shown in Figure 4.14(a). Each line corresponds to a different initial starting condition. The dimension of X_v is equal to 4 for each non-leaf vertex v and equal to 8 for each leaf vertex v. (b) Same type of plot as in (a), but here, the dimension of X_v is equal to 1 for each non-leaf vertex v.



(a)

(b)

Figure 4.16. (a) Shows the absolute error between the true covariance in Figure 4.14(a) and an approximate solution generated after 1000 iterations of (4.57). In this case, the dimension of X_v is equal to 4 for each non-leaf vertex v and equal to 8 for each leaf vertex v. (b) Same type of plot as in (a), but here, the dimension of X_v is equal to 1 for each non-leaf vertex v.

Chapter 5

Conclusions and Future Research Directions

THIS thesis provides a detailed study of multiscale models, their independence properties, and the multiscale realization problem. This study reveals that a thorough understanding of the conditional independencies exhibited by multiscale models plays an important role in the development of efficient multiscale realization algorithms, a statement which is supported by the various contributions summarized in Section 5.1. While several facets of multiscale models and the multiscale realization problem have been explored, there are still a number of unanswered questions. Section 5.2 addresses some of these open questions by suggesting appropriate extensions to the realization framework developed in this thesis.

5.1 Summary of Contributions

The contributions of this thesis may be broadly categorized as follows:

- Study of a novel type of Markov property – marginalization-invariant Markovianity.
- Development of a sequential procedure for solving the exact multiscale realization problem (with and without augmented states).
- Development of a graph-theoretic framework for enumerating the conditional independencies of multiscale models, as well as more complex graphical models.
- Formulation of an approximate multiscale realization problem for which the degree of approximation is tied to the constraints imposed by the marginalization-invariant Markov property.
- Application of the EM algorithm to the approximate multiscale realization problem.

The first contribution of this thesis is the most fundamental, and perhaps the most important, since the remaining contributions are directly tied to or rely upon the marginalization-invariant Markov property. This property is defined in Chapter 2 in terms of so-called boundary sets which depend on the choice of the marginalization constraint set M and the sets associated with the reduced-order global Markov property. The marginalization-invariant Markov property is important because it provides a set of sufficient conditions which solutions to the multiscale realization problem must satisfy. These sufficient conditions, however, are not unique for a given multiscale model and depend on the ordering chosen for the non-leaf vertices of the tree.

The structure of these sufficient conditions leads directly to the second contribution of this thesis – a sequential procedure for the exact multiscale realization problem. In the second part of Chapter 2, we demonstrate that the constraints imposed by the marginalization-invariant Markov property may be specially ordered when the marginalization constraint set M contains only leaf vertices, and we use this fact to derive a sequential realization procedure. In the cases where M contains non-leaf vertices, we demonstrate that a sequential realization procedure is also possible if additional design vectors are introduced into the problem.

The fact that the marginalization-invariant Markov property is not unique for a given tree implies that many different sequential realization procedures may be devised. Namely, a chosen ordering on the non-leaf vertices of the tree leads to a particular set of sufficient conditions and a specific sequential realization procedure based on these conditions. By changing the ordering, a different realization procedure is obtained. For example, in Section 2.8.3 we demonstrate that a bottom-up ordering[1] leads to a procedure which resembles the scale-recursive algorithm discussed in [38]. Similarly, choosing a top-down ordering leads to a procedure like the one proposed in [49]. However, since there are combinatorially many such procedures, it is not always clear which one is best for a given problem; Section 5.2.1 provides some guidance on how to think about this problem from a graph-theoretic perspective.

Even though the marginalization-invariant Markov property is defined and studied in Chapter 2, its significance for the multiscale realization problem is not proven until Chapter 3. To help prove its significance, the first part of Chapter 3 develops an interesting graph-theoretic framework for enumerating the conditional independencies of multiscale models, as well as more complex graphical models. Using this framework, we then show that the marginalization-invariant Markov property provides a set of sufficient conditions for solutions to the multiscale realization problem.

An additional benefit of the graph-theoretic framework introduced in Chapter 3 is that it leads to an important decomposition of the KL divergence between a density p and its projection p^T onto a tree. In the second part of Chapter 3, we formulate an approximate version of the multiscale realization problem which uses the KL divergence as a measure of approximation, and we show that there exists an upper bound on this criterion which possesses several important and useful properties. In particular, we prove that this upper bound may be additively decomposed into terms which measure the constraints imposed by the marginalization-invariant Markov property.

In Chapter 4, we continue our discussion of the approximate multiscale realization problem by proposing a specific iterative procedure for solving it. We view this procedure from several different perspectives, and in particular, we demonstrate that it seeks solutions which simultaneously tradeoff the desired marginal constraint with the constraints imposed by the global Markov property – the same two constraints satisfied by the marginalization-invariant Markov property. We subsequently demonstrate how to apply this iterative procedure to parameterized approximate realization problems, and we demonstrate that such a procedure is in fact equivalent to the EM algorithm. In the final part of Chapter 4, we apply this iterative procedure to the Gaussian multiscale realization problem, and we provide several interesting examples which validate its performance in practice.

[1] See Section 2.6.2 for a review of bottom-up and top-down orderings.

5.2 Suggestions for Future Research

This section focuses on several questions which are raised in this thesis but not answered, as well as several questions not raised in this thesis but for which the framework developed here is applicable.

5.2.1 Conditional Independence and Minimality

In Chapter 3, Theorem 3.3 suggests a set of sufficient conditions which solutions to the multiscale realization problem must satisfy – conditions which are stated in terms of specific triangulated supergraphs of trees. In this section, we consider an equivalent set of conditions which holds for more general graphical models, and we demonstrate that the complexity of these conditions may be analyzed from a graph-theoretic perspective. Specifically, we show how so-called *minimal triangulations*, as well as the notions of clique extensions and neighborhood separators previously introduced in Chapter 3, are important components for understanding complexity. Our discussion suggests several interesting research questions which should be addressed in order to better understand the multiscale realization problem as well as more complex realization problems. We conjecture that this graph-theoretic perspective may ultimately be useful for finding minimal-complexity realization algorithms for a broader class of graphical models.

Minimal Triangulations

Consider now a generalization of the sufficient conditions stated in Theorem 3.3 which holds for more complex graphical models. Suppose a triangulated graph $\mathcal{G} = (V, E)$, a subset of vertices $M \subset V$, and a target density $p^*(x_M)$ are given. In this scenario, the exact realization problem consists of identifying a density q which factors according to $\mathcal{G}$ and satisfies $q(x_M) = p^*(x_M)$. One method for finding solutions is to identify densities p which satisfy the following set of conditions:

> <u>Sufficient Conditions $\mathcal{S}$</u>
> Given a triangulated graph $\mathcal{G} = (V, E)$ and $M \subset V$, let $\mathcal{G}'$ be a triangulated supergraph of $\mathcal{G}$ which contains a clique equal to M. A density p generates a solution $q = p_{\mathcal{G}}$ to the realization problem if $p(x_M) = p^*(x_M)$ and $p_{\mathcal{G}'} = p_{\mathcal{G}}$.

The result trivially follows from the sequence of equalities $p^*(x_M) = p(x_M) = p_{\mathcal{G}'}(x_M) = p_{\mathcal{G}}(x_M) = q(x_M)$.

The conditions stated above are important because they demonstrate which conditional independencies are relevant for the realization problem – namely the conditional independencies which ensure that the equality $p_{\mathcal{G}'} = p_{\mathcal{G}}$ holds. Since the complexity of these conditional independencies is directly related to the number of additional edges contained in $\mathcal{G}'$ but not in $\mathcal{G}$, it is important that $\mathcal{G}'$ have the fewest number of edges possible. Of course, $\mathcal{G}'$ must be both triangulated and have a clique equal to M, and as such, the number of additional edges can be quite large for some types of graphs and some choices of M.

More formally, the preceding requirements on $\mathcal{G}'$ may be stated in terms of the graph-theoretic notion called minimal triangulation. Given a non-triangulated graph $\mathcal{H} = (U, F)$, a minimal triangulation $\mathcal{H}' = (U, F \cup F')$ of $\mathcal{H}$ is one where every edge in F' is required in order for $\mathcal{H}'$ to be triangulated, *i.e.* if any edge or set of edges is removed from F', the resulting graph is not triangulated. In order for a graph $\mathcal{G}'$ to satisfy conditions $\mathcal{S}$, the graph $\mathcal{G}'$ should be a minimal triangulation of $\mathcal{G} \cup K_M$, *i.e.* the graph formed by the union of the edges contained in $\mathcal{G}$ and the complete graph on M. Such a triangulation provides a graph $\mathcal{G}'$ which not only satisfies the conditions $\mathcal{S}$ but is also minimal in the sense that removing any edges results in a graph which does not satisfy conditions $\mathcal{S}$. It is important to note, however, that minimal triangulations are not unique: some minimal triangulations can contain more edges than others. Minimal triangulations with the fewest number of edges are called minimum triangulations.

For general graphs, finding minimal triangulations can be somewhat challenging, and finding a minimum triangulation is NP-hard. These types of problems have been studied extensively in the graph theory literature, and there are several techniques for finding minimal triangulations [7,46,82]. For the multiscale realization problem, we conjecture that the modified elimination game in fact generates minimal triangulations for tree-structured graphs as long as a leaf-last vertex ordering is chosen. If this conjecture is true, it then suggests that the constraints imposed by marginalization-invariant Markov property are irreducible. Furthermore, it raises the question of which classes of graphs and which vertex orderings lead to minimal triangulations when the modified elimination game is used.

Conditional Independence Constraints with Minimal Redundancy

While minimal triangulations are important for understanding the complexity of realization algorithms, there is another important aspect of complexity which must be addressed – redundant conditional independence constraints. As an example of redundant constraints, recall from Chapter 2 that the global Markov property imposes a set of overlapping conditional independence constraints. We subsequently showed that this overlap may be diminished by considering the constraints imposed by the reduced-order global Markov property. Essentially, the reduced-order global Markov property removes constraints which are enforced more than once and are therefore redundant. From the perspective of the realization problem, it is important to remove all redundant constraints since this redundancy impacts the computational efficiency of any algorithm which attempts to enforce these constraints.

We conjecture that the graph-theoretic concepts introduced in Chapter 3, specifically clique extensions and neighborhood separators, provide the tools necessary for eliminating redundancy in a set of conditional independence constraints. In particular, we argue that a subgraph $\mathcal{H}$ induced by the neighborhood of a neighborhood separator represents the most complex non-complete graph for which all implied Markov properties may be compactly stated without redundancy. For example, consider the graph $\mathcal{G}$ shown in Figure 5.1 which contains the subgraph $\mathcal{H}$ induced by the neighborhood of the neighborhood separator $S = \{u, v\}$. In this example, the set of vectors X_r, X_s, X_t, X_u, X_v are Markov with respect to $\mathcal{H}$ if and only if they satisfy the following constraint,

$$X_r \perp X_s \perp X_t \,|\, (X_u, X_v) \,. \tag{5.1}$$

Even though the preceding constraint may be represented in a number of equivalent ways,[2] no other representation of this constraint provides additional benefit over the one provided in (5.1). Based on this, we conjecture that neighborhood separators are the fundamental building blocks for studying conditional independencies.

Besides neighborhood separators, we conjecture that clique extensions provide the most natural method for enumerating the conditional independencies implied by graphs with multiple neighborhood separators. For example, in Chapter 3 we used clique extensions to compare the conditional

[2] For example, the two constraints $X_r \perp X_s \,|\, (X_u, X_v)$ and $X_t \perp (X_r, X_s) \,|\, (X_u, X_v)$ taken together are equivalent to the single constraint $X_r \perp X_s \perp X_t \,|\, (X_u, X_v)$.

independencies exhibited by two densities $p_{\mathcal{G}}$ and $p_{\mathcal{G}'}$, where $\mathcal{G}$ and $\mathcal{G}'$ are triangulated and $\mathcal{G}'$ is a supergraph of $\mathcal{G}$. We first showed that a sequence of clique extensions $\mathcal{G} = \mathcal{G}_0, \mathcal{G}_1, \ldots, \mathcal{G}_n = \mathcal{G}'$ always exists for such choices of $\mathcal{G}$ and $\mathcal{G}'$. Then, using this sequence along with Theorem 3.7, we showed how the conditional independencies exhibited by $p_{\mathcal{G}}$ but not $p_{\mathcal{G}'}$ can be easily listed by examining the maximal cliques generated in this sequence. We conjecture that all such graphs $\mathcal{G}$ and $\mathcal{G}'$ may be decomposed in terms of clique extensions such that a non-redundant set of constraints is always obtained.

As an example, consider the sequence of clique extensions $\mathcal{G} = \mathcal{G}_0, \mathcal{G}_1, \mathcal{G}_2 = \mathcal{G}'$ shown in Figure 5.2(a), (b), and (c) respectively. Using this sequence, we can determine a non-redundant set of conditional independencies implied by the graph $\mathcal{G}$ by examining the two maximal cliques $\{1, 2, 3, 4\}$ and $\{0, 1, 2, 3, 4\}$ formed in this sequence. Doing so suggests that a density $p_{\mathcal{G}}$ exhibits the following conditional independencies

$$(X_1, X_2) \perp X_4 | X_3 \tag{5.2a}$$

$$X_0 \perp (X_3, X_4) \,|\, (X_1, X_2)\,. \tag{5.2b}$$

Alternatively, if we examine the graph $\mathcal{G}$ directly, it is immediately clear that $p_{\mathcal{G}}$ satisfies the following conditional independencies,

$$(X_0, X_1, X_2) \perp X_4 | X_3 \tag{5.3a}$$

$$X_0 \perp (X_3, X_4) \,|\, (X_1, X_2)\,, \tag{5.3b}$$

but it is not immediately obvious that random vector X_0 may be removed from the conditional independence statement in (5.3a), as in (5.2a), without changing the Markov properties implied by both statements in (5.3). This example illustrates the utility of clique extensions for minimizing redundancy in a set of conditional independence constraints.

This same example also illustrates the important point that not all clique extensions result in a non-redundant set of conditional independencies. For example, the graphs $\mathcal{G}, \mathcal{G}'$ form a sequence of clique extensions since $\mathcal{G}'$ is the complete graph. However, this sequence results in the creation of the maximal clique $\{0, 1, 2, 3, 4\}$ which requires us to examine the conditional independencies of the entire graph $\mathcal{G}$, and as previously stated, it can be difficult to determine a non-redundant set of conditional independencies for an entire graph at once. This underscores the importance of the particular choice of clique extension. We conjecture that this issue is not a problem if a chordal sequence of clique extensions is chosen. Since a chordal sequence adds edges one-at-a-time, we conjecture that a non-redundant set of constraints is guaranteed in this situation.

Minimum-Complexity Realization Algorithms

While the graph-theoretic ideas of minimal triangulations, neighborhood separators, and clique extensions provide an interesting methodology for listing conditional independencies, we are ultimately interested in the realization algorithms which result from this methodology. For example, in the case of the multiscale realization problem, we demonstrated that the constraints imposed by the marginalization-invariant Markov property may be ordered (when M contains only leaf vertices) so that non-leaf vectors X_v may be designed in a sequential fashion. We conjecture that any non-redundant set of constraints, such as that generated by a sequence of clique extensions, immediately suggests a sequential procedure for a much broader class of graphical models. For example,

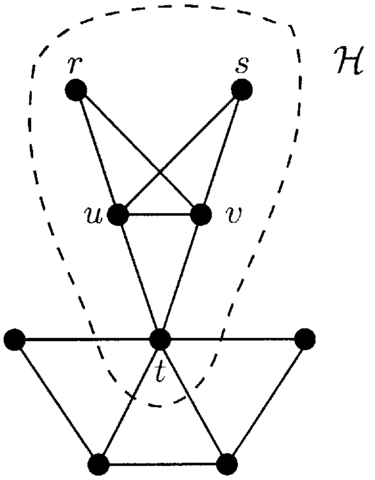


Figure 5.1. A graph $\mathcal{G}$ where the subgraph $\mathcal{H}$ is induced by the neighborhood of the neighborhood separator $S = \{u, v\}$.

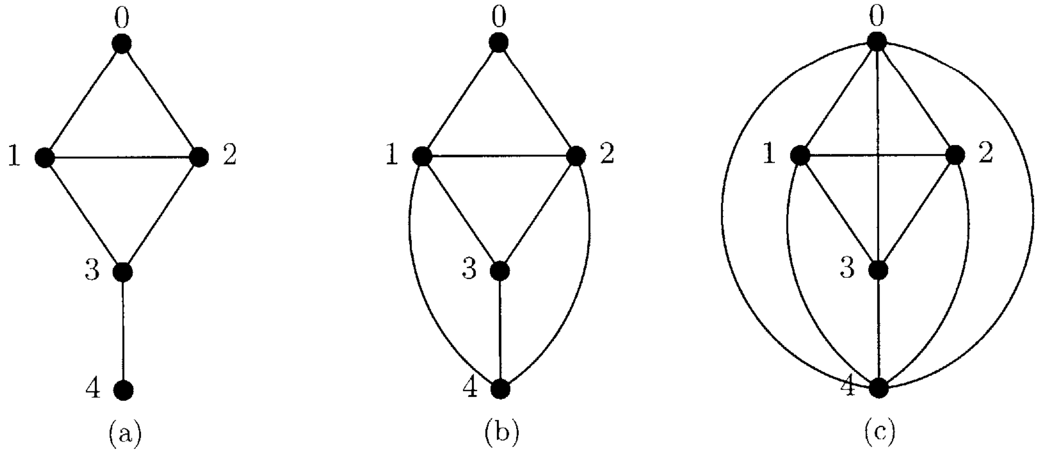


Figure 5.2. Illustration of a sequence of clique extensions $\mathcal{G}_0$, $\mathcal{G}_1$, and $\mathcal{G}_2$, where $\mathcal{G}_0$ is shown in (a), $\mathcal{G}_1$ in (b), and $\mathcal{G}_2$ in (c).

Chapter 3 proves that a sequential procedure is available for graphical models whose underlying structure is given by an augmented graph.

Another issue related to sequential realization concerns which sets of constraints lead to the most efficient realization procedures. As previously discussed, one set of constraints may be more complex than another due to the minimal triangulation and the clique extensions which are chosen. In addition, the dimensionality of each random vector X_v is an important aspect of complexity, and therefore, it is essential that dimensionality be accounted for when deciding between different sets of constraints. One method for attacking this problem is to formulate the search for the optimal set of constraints in terms of the search for an optimal sequence of edges. Specifically, we can treat this problem as a graph optimization problem, where weights are associated with edges in the graph and provide an appropriate measure of complexity. Such an optimization problem could be NP-hard for general graphs, but it may be possible to derive guaranteed bounds on complexity for particular subclasses of graphs.

■ 5.2.2 Measuring Conditional Independence

While the previous section focused on how to derive a minimal set of sufficient conditions for solutions to a realization problem, this section focuses on how to measure conditional independencies. This is an important aspect of any realization problem since it is not always possible to satisfy a set of constraints exactly, as discussed earlier in the context of the approximate multiscale realization problem. In this thesis, we focused exclusively on the measure of approximation provided by the KL divergence; however, there are other measures of note which may prove useful for other types of realization problems.

One possible measure is the one provided by canonical correlations, as used in [49] for the Gaussian multiscale realization problem. Canonical correlations can be used to measure the conditional correlation between two random vectors X_1 and X_2, conditioned on some linear function of X_1, and this measure can then be optimized to find the best linear function which conditionally decorrelates X_1 and X_2. It can be shown, however, that the exact same linear function is obtained by optimizing the conditional KL divergence. Consequently, we believe that the KL divergence is a natural generalization of any measure based on canonical correlations.

A second possibility for the Gaussian multiscale realization problem is the predictive efficiency measure proposed in [38]. This measure is computationally simpler to optimize than canonical correlations, and it can be interpreted as a measure of estimation error. The difficulty with this measure (as well as canonical correlations) is that it can only be used to conditionally decorrelate two random vectors at a time. Consequently, this measure does not naturally lend itself to the problem of simultaneously decorrelating more than two random vectors, as required in the multiscale realization problem for example. We conjecture that predictive efficiency represents a specific upper bound on the KL divergence. By viewing predictive efficiency as an approximation to the KL divergence, it should be possible to generalize this measure so that more complex conditional independencies can be handled. Ultimately, the hope would be that this measure provides a notion of approximation which is relevant for a particular type of realization problem and for which its optimization is simpler than the optimization required for the KL divergence.

In addition to canonical correlations, predictive efficiency, and KL divergence, there are other measures of conditional independence which may be of possible interest. These include the multi-information function [100], the kernel independent component analysis proposed in [3], and various measures used in signal processing such as the Hellinger, Chernoff, and Bhattacharyya distances [5].

The kernel independent component analysis, for example, provides a means of measuring conditional independence when densities are non-Gaussian; such a measure could be useful for solving realization problems involving non-Gaussian target densities $p^*(x_M)$.

In considering other measures of conditional independence, it is also important that such a measure be decomposable. As demonstrated in Proposition 3.24, the KL divergence can be decomposed into a sum of terms, where each term measures a single independence constraint. Ideally, any chosen measure of conditional independence should be decomposable in a similar manner, so that the contribution of each independence constraint to overall approximation quality may be easily identified and optimized.

■ 5.2.3 Iterative Methods for Solving the Approximate Multiscale Realization Problem

In this thesis, we use the EM algorithm as a method for solving the approximate multiscale realization problem, but there are other possibilities which may be of use for this problem as well as more general realization problems. In this section, we suggest two possibilities that are based on Propositions 3.23 and 3.24. Recall from Proposition 3.23 that the multiscale realization problem may be solved by minimizing an upper bound – a bound which Proposition 3.24 shows to be decomposable into a sum of simpler terms. Both of the methods suggested here involve minimizing this upper bound by using its additive structure.

A Dynamic Programming Approach

First consider a dynamic programming approach to minimizing the upper bound provided in Proposition 3.23. This approach relies on the fact that the constraints imposed by the marginalization-invariant Markov property may be ordered so that vectors X_v are designed sequentially. This implies that the terms of the additive decomposition in Proposition 3.24 may be ordered so that each term in the decomposition can be individually optimized with respect to a single design vector X_v. However, even though each term can be treated in such a myopic way, other terms in the additive decomposition also depend on the choice of X_v, and consequently, we must account for the contributions of these other terms to the total cost.

This type of sequential dependence on previous decisions is reminiscent of dynamic programming problems [8], and therefore, we believe that it may be useful to formulate the multiscale realization problem as a sequential optimization problem. As an example of what we mean, consider the tree shown in Figure 5.3 where $M = \{3, 4, 5, 6\}$ contains all leaf vertices. Using Proposition 3.24, the function $D\left(p(x)||p^T(x)\right)$ to be minimized may be written as follows,

$$\begin{aligned} D\left(p(x)||p^T(x)\right) = & \, D\left(p(x_1, x_3, x_4, x_5, x_6)||p(x_3|x_1)p(x_4|x_1)p(x_5, x_6|x_1)p(x_1)\right) + \\ & D\left(p(x_1, x_2, x_5, x_6)||p(x_1|x_2)p(x_5|x_2)p(x_6|x_2)p(x_2)\right) + \\ & D\left(p(x_0, x_1, x_2)||p(x_1|x_0)p(x_2|x_0)p(x_0)\right). \end{aligned} \tag{5.4}$$

In the first term, random vectors X_3, X_4, X_5, X_6 are defined by the joint density $p^*(x_M)$, and therefore, the only degree of freedom is the choice for the conditional density $p(x_1|x_3, x_4, x_5, x_6)$ which defines X_1. In choosing X_1, however, we must account for its influence on the two remaining terms in (5.4) which also depend on X_1. Assuming that random vector X_1 can be designed appropriately, the second term in (5.4) is then only a function of X_2. When optimizing this term, though, we must recognize the fact that the third term also depends on X_2. Finally, given choices for X_1 and X_2, the third term may be optimized with respect to X_0.

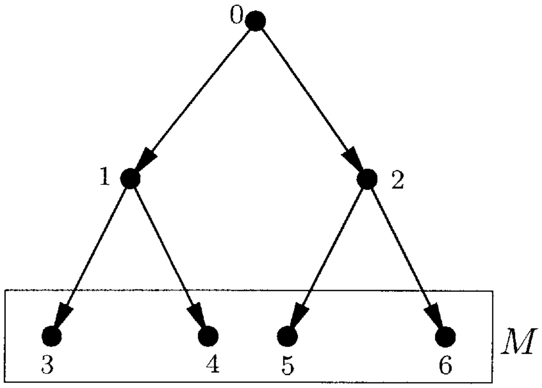


Figure 5.3. The tree considered in the discussion of a dynamic programming approach to the multiscale realization problem.

We conjecture that dynamic programming may be a useful way to deal with the type of interdependency displayed in the preceding example. However, this type of problem could prove difficult to solve exactly, and therefore, it may be useful to consider approximate iterative dynamic programming methods. Since the scale-recursive realization algorithm proposed in [38] is essentially a greedy version of what we propose here, *i.e.* each term in (5.4) is optimized separately, we believe that performing a few iterations of an approximate dynamic programming algorithm will necessarily lead to significantly better multiscale models.

A Conjugate Gradient Approach

Besides the dynamic programming approach mentioned above, it is also possible to apply other types of optimization techniques to the multiscale realization problem. The method of conjugate gradients [9, 94] is one optimization technique which may prove useful due to its excellent convergence properties for a broad array of problems. For example, recent work, which considers conjugate gradients in the context of the realization problem, suggests an interesting relationship between the EM algorithm and a so-called expectation-conjugate-gradient algorithm [92].

The novelty of the approach suggested here is that it relies on the upper bound given in Proposition 3.23, which as far as we can tell has not been considered in the realization literature. Since this bound may be additively decomposed in terms of more localized functions, we conjecture that the gradient of this bound (with respect to a particular parametrization) may be decomposable as well. Since the method of conjugate gradients only requires a cost function and its gradient to be calculated, this method of optimization may represent a viable alternative to the EM algorithm.

■ 5.2.4 Partial Specifications

An interesting and important generalization of the problem framework considered in this thesis – and one that has close connections to problems such as the so-called covariance completion problem – is the following. Suppose that instead of having a single set of vertices M and target density $p^*(x_M)$ we have several possibly overlapping sets of vertices $M_1, M_2, \ldots$ and associated target densities $p^*(x_{M_1}), p^*(x_{M_2}), \ldots$. In this case, the question of determining multiscale or other graphical model realizations has a feature not found in the problem studied in this thesis. In particular, there is the issue of the *compatibility* of a set of target densities, *i.e.* whether any density over the set of vertices $M_1 \cup M_2 \cup \ldots$ could have these specified marginal densities. Indeed, this question represents a generalization of the question of whether a partially specified covariance matrix has any positive definite completion.

The motivation for this generalization comes from the same source as that for covariance completion; namely, available measurement data may only provide estimates of these densities over subsets of variables. Developing methods for determining exact realizations, of course, would have to deal with whether the given target densities are compatible. In principle, however, the problem of determining models that approximate each of these target densities would not require such compatibility but would require the use of a cost function that separately combines the error measures for each of the target densities.

Appendix A

Proofs for Chapter 2

A.1 Proof of Proposition 2.1

We note that the proof of the following proposition is a simple adaptation of the proof given in [38] for multiscale autoregressive models.

Proposition 2.1 (Equivalence of Internal and Locally Internal Multiscale Models). *A multiscale model is internal if and only if it is locally internal.*

Proof. As previously discussed in Section 2.3.3, any locally internal tree-indexed process is also an internal tree-indexed process; therefore, this fact also holds for multiscale models.

To prove the other direction, assume that $(X, \mathcal{G}_{\preceq})$ is an internal multiscale model. We show that for any non-leaf vertex v, the vector X_v is equal to $E[X_v|X_{\chi(v)}]$, thereby implying that X_v can be written as a deterministic function of the process indexed by the children of v, *i.e.* $X_{\chi(v)}$. To begin, we can always write

$$X_v = E[X_v|X_{\chi(v)}] + \tilde{X}_v, \tag{A.1}$$

for some $\tilde{X}_v$. The result follows if we can show that $\tilde{X}_v$ is equal to zero. Using (A.1), note that $\tilde{X}_v$ is a deterministic function of X_v and $X_{\chi(v)}$, and therefore, we can write

$$\tilde{X}_v = E[\tilde{X}_v|X_v, X_{\chi(v)}]. \tag{A.2}$$

Since $(X, \mathcal{G}_{\preceq})$ is internal, we know that both X_v and $X_{\chi(v)}$ are functions of the process X_{L_v}, and using (A.2), $\tilde{X}_v$ is also some function of X_{L_v}. The global Markov property of multiscale models then implies that $\tilde{X}_v$ (a function of X_{L_v}) and X_v are conditionally independent given $X_{\chi(v)}$, and consequently,

$$\tilde{X}_v = E[\tilde{X}_v|X_v, X_{\chi(v)}] = E[\tilde{X}_v|X_{\chi(v)}] = E[X_v - E[X_v|X_{\chi(v)}]|X_{\chi(v)}] = 0. \tag{A.3}$$

The final equality follows from the definition of $\tilde{X}_v$ in (A.1). ■

A.2 Proof of Proposition 2.2 and Theorem 2.2

Recall from Section 2.5 that for all non-leaf vertices v, with $v \neq v_i$, we use the notation $S_v^{v_i}$ to represent the unique set $S \in \mathcal{S}_v$ that contains v_i. Recall also that such a set $S_v^{v_i}$ can take one of two forms,

$$\begin{aligned}
&\text{Type 1: } S_v^{v_i} = \bar{S}_u, \text{ where } u \in \chi(v) \text{ such that } v_i \succeq u,\\
&\text{Type 2: } S_v^{v_i} = S_v^c \cup \{v\}.
\end{aligned}$$

We henceforth refer to these two different forms as Type 1 and Type 2.

To prove Proposition 2.2 and Theorem 2.2, it is necessary to provide several intermediate results. The first of these provides an important characterization of the intersection of any two sets $S_v^{v_i}$ and $S_{v'}^{v_i}$.

Lemma A.1 (Intersection of Subtrees).
Let v and v' be distinct non-leaf vertices of a graph $\mathcal{G}_{\preceq}$. Suppose $S \in \mathcal{S}_v$, $S' \in \mathcal{S}_{v'}$, and both S and S' contain a common element v^*.

(1) $v \notin S \cap S'$ if and only if $S' \subset S$.

(2) Both $v \in S \cap S'$ and $v' \in S \cap S'$ only in the following cases:

(i) S is Type 1, S' is Type 2, and $v' \in S$.

(ii) S is Type 2, S' is Type 1, and $v \in S'$.

(iii) S and S' are Type 2, and v and v' are not comparable with the partial order $\preceq$.

Proof. Before proving either result, we consider all possible combinations of S and S'. We present a graphical proof by considering the exhaustive set of examples shown in Figures A.1 and A.2. Consider the four possible combinations for S and S':

- S is Type 1 and S' is Type 1:

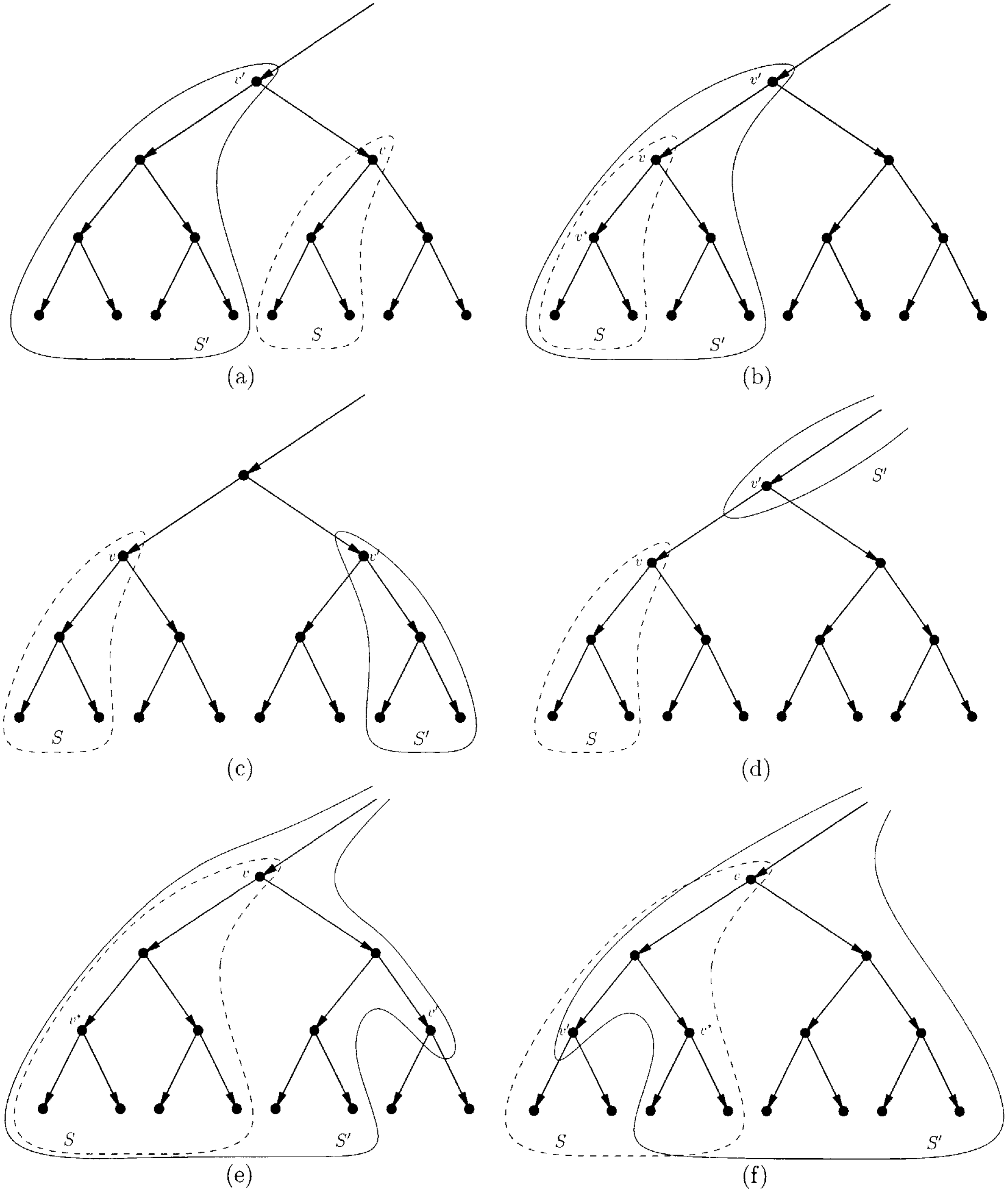
$$\begin{aligned}
v \succ v',\ v \notin S' &\Longrightarrow v^* \notin S \cap S' = \emptyset \\
v \succ v',\ v \in S' &\Longrightarrow S' \not\subset S,\ v \in S \cap S',\ v' \notin S \cap S' \\
v' \succ v,\ v' \notin S &\Longrightarrow v^* \notin S \cap S' = \emptyset \\
v' \succ v,\ v' \in S &\Longrightarrow S' \subset S,\ v \notin S \cap S',\ v' \in S \cap S' \\
v \not\succ v',\ v' \not\succ v &\Longrightarrow v^* \notin S \cap S' = \emptyset
\end{aligned}$$

The first two statements follow from the illustrations shown in Figures A.1(a) and (b) respectively. The third and fourth statements follow from Figures A.1(a) and (b) respectively by reversing the vertices v and v' and their corresponding sets S and S'. The fifth statement follows from Figure A.1(c).

- S is Type 1 and S' is Type 2:

$$\begin{aligned}
v \succ v' &\Longrightarrow v^* \notin S \cap S' = \emptyset \\
v' \succ v,\ v' \notin S &\Longrightarrow S' \not\subset S,\ v \in S \cap S',\ v' \notin S \cap S' \\
v' \succ v,\ v' \in S &\Longrightarrow S' \not\subset S,\ v \in S \cap S',\ v' \in S \cap S' \\
v \not\succ v',\ v' \not\succ v &\Longrightarrow S' \not\subset S,\ v \in S \cap S',\ v' \notin S \cap S'
\end{aligned}$$

The first three statements follow from Figures A.1(d), (e), and (f) respectively. The final statement follows from Figure A.2(a).



(a) (b) (c) (d) (e) (f)

Figure A.1: Set of examples used to graphically prove the results in Lemma A.1.

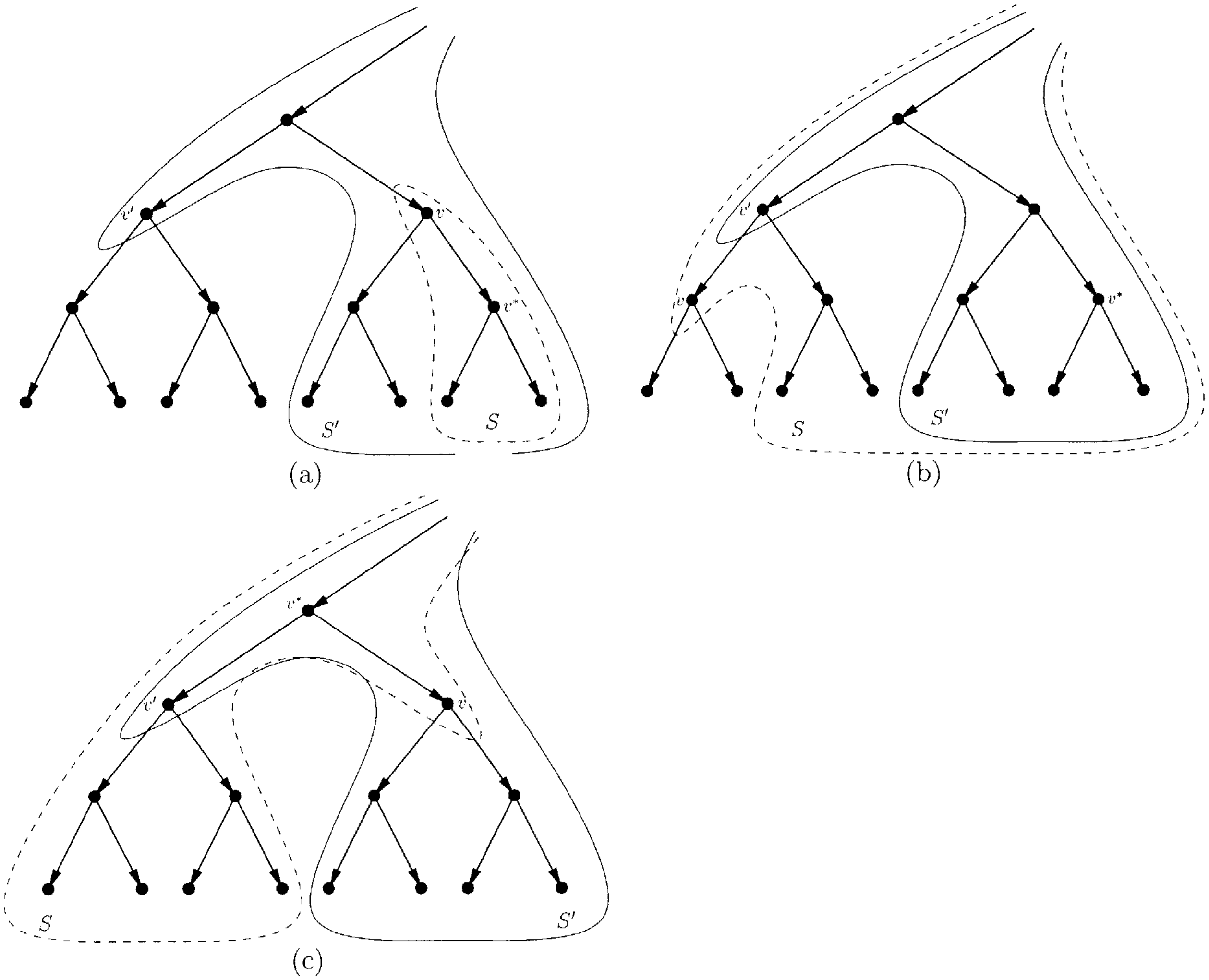


Figure A.2: Set of examples used to graphically prove the results in Lemma A.1.

- S is Type 2 and S' is Type 1:

$$\begin{aligned} v \succ v',\ v \notin S' &\Longrightarrow S' \subset S,\ v \notin S \cap S',\ v' \in S \cap S' \\ v \succ v',\ v \in S' &\Longrightarrow S' \not\subset S,\ v \in S \cap S',\ v' \in S \cap S' \\ v' \succ v &\Longrightarrow v^* \notin S \cap S' = \emptyset \\ v \not\succ v',\ v' \not\succ v &\Longrightarrow S' \subset S,\ v \notin S \cap S',\ v' \in S \cap S' \end{aligned}$$

The first three statements follow by reversing the role of v and v' in Figures A.1(e), (f), and (d) respectively, and the final statement follows by reversing the role of v and v' in Figure A.2(a).

- S is Type 2 and S' is Type 2:

$$\begin{aligned} v \succ v' &\Longrightarrow S' \subset S,\ v \notin S \cap S',\ v' \in S \cap S' \\ v' \succ v &\Longrightarrow S' \not\subset S,\ v \in S \cap S',\ v' \notin S \cap S' \\ v \not\succ v',\ v' \not\succ v &\Longrightarrow S' \not\subset S,\ v \in S \cap S',\ v' \in S \cap S' \end{aligned}$$

The first statement follows from the illustration shown in Figure A.2(b). The second statement follows from Figure A.2(b) by reversing v and v'. The final statement follows from Figure A.2(c).

(1) To prove the first result, we must show that in all cases where S and S' contain a common element v^*, the statements $v \notin S \cap S'$ and $S' \subset S$ are equivalent. Examining the cases discussed above shows that this is in fact true.

(2) Examining the different combinations of S and S', we see that $v \in S \cap S'$ and $v' \in S \cap S'$ in only the three scenarios which were listed. ■

Recall the definition of the boundary set of vertex $v_i \in (v_2, \ldots, v_m)$,

$$B_{v_i} = \bigcap_{v < v_i} S_v^{v_i}, \tag{A.4}$$

and the corresponding set T_{v_i},

$$T_{v_i} \triangleq \{v | v \in B_{v_i}, v < v_i\}, \tag{A.5}$$

containing vertices in the boundary B_{v_i} that precede vertex v_i in the ordering. An immediate consequence of Lemma A.1 is the following corollary that provides a characterization of the boundary B_{v_i} which is similar to (A.4) but ignores sets $S_v^{v_i}$ that do not influence B_{v_i}.

Corollary A.1 (Characterization of the Boundary Sets).
The boundary set B_{v_i} can be written as the intersection over the sets $S_v^{v_i}$ for all $v \in T_{v_i}$,

$$B_{v_i} = \bigcap_{v \in T_{v_i}} S_v^{v_i}. \tag{A.6}$$

Proof. Assuming that $T_{v_i} \neq \{v_1, \ldots, v_{i-1}\}$, choose any element $v \in \{v_1, \ldots, v_{i-1}\}$, $v \notin T_{v_i}$. Since $v \notin T_{v_i}$, we know that $v \notin B_{v_i}$, and therefore, there must exist at least one vertex $v' < v_i$ such that $v \notin S_v^{v_i} \cap S_{v'}^{v_i}$. Using the first part of Lemma A.1, this implies that $S_{v'}^{v_i} \subset S_v^{v_i}$, and consequently, the set $S_v^{v_i}$ does not influence the boundary B_{v_i}. The vertex v can be safely removed from the definition of B_{v_i}. ■

Another important consequence of Lemma A.1 is the following characterization of the elements of the set T_{v_i} and the associated sets $S_t^{v_i}$ for $t \in T_{v_i}$.

Lemma A.2 (Characterization of T_{v_i}).

(1) The set T_{v_i} is non-empty.

(2) If T_{v_i} contains at least two elements, then there exists at most one element $t^* \in T_{v_i}$ which is comparable to $t \in T_{v_i}$, $t \neq t^*$, with respect to the partial order $\preceq$. If such a t^* exists then $t \in S_{t^*}^{v_i}$ for all $t \in T_{v_i}$.

(3) The set $S_{t^*}^{v_i}$ is Type 1. For any $t \in T_{v_i}$, $t \neq t^*$, the set $S_t^{v_i}$ is Type 2.

Proof.

(1) By Corollary A.1, if $T_{v_i} = \emptyset$, then $B_{v_i} = \emptyset$ which is a contradiction since B_{v_i} by definition contains at least v_i.

(2-3) Let t and t^* be distinct elements of T_{v_i}, and consider the sets $S_t^{v_i}$ and $S_{t^*}^{v_i}$. First, suppose that $S_t^{v_i}$ and $S_{t^*}^{v_i}$ are both Type 1. The second part of Lemma A.1 implies that either $t \notin S_t^{v_i} \cap S_{t^*}^{v_i}$ or $t^* \notin S_t^{v_i} \cap S_{t^*}^{v_i}$. This in turn implies that either $t \notin T_{v_i}$ or $t^* \notin T_{v_i}$, which is a contradiction. This indicates that there can be at most one set $S_{t^*}^{v_i}$, $t^* \in T_{v_i}$, of Type 1.

Suppose now that $S_{t^*}^{v_i}$ is Type 1 and $S_t^{v_i}$ is Type 2 for all $t \neq t^*$. Using the second part of Lemma A.1, the only way we can have $t \in S_t^{v_i} \cap S_{t^*}^{v_i}$ and $t^* \in S_t^{v_i} \cap S_{t^*}^{v_i}$ is if $t \in S_{t^*}^{v_i}$. Therefore, $t \in S_{t^*}^{v_i}$ for all $t \in T_{v_i}$, and $S_t^{v_i}$ is Type 2 for all $t \neq t^*$.

Finally, suppose $S_t^{v_i}$ and $S_{t'}^{v_i}$, $t \neq t'$, are Type 2. The second part of Lemma A.1 indicates that t and t' cannot be comparable since otherwise we arrive at a contradiction. Thus, all sets $S_t^{v_i}$, $t \neq t^*$ are Type 2, and all elements of T_{v_i} not equal to t^* are incomparable with respect to $\preceq$. ■

Lemma A.2 allows the characterization of B_{v_i} in Corollary A.1 to be strengthened.

Corollary A.2 (Characterization of the Boundary Sets).
If a t^* as described in Lemma A.2 exists, then let u^* be the vertex such that $S_{t^*}^{v_i} = \bar{S}_{u^*}$. The boundary set B_{v_i} may be written as

$$B_{v_i} = \left[\bigcap_{t \in T_{v_i} - \{t^*\}} (S_t^c \cup \{t\}) \right] \cap \bar{S}_{u^*}. \tag{A.7}$$

Otherwise, if no such t^* exists then

$$B_{v_i} = \bigcap_{t \in T_{v_i}} (S_t^c \cup \{t\}). \tag{A.8}$$

Proof. The result follows directly from Corollary A.1 and Lemma A.2. ■

Finally, we need the following lemma which provides an important characterization of the reduced-order sets.

Lemma A.3 (Reduced-Order Sets).
Let $v_i \in (v_2, \ldots, v_m)$ and $v^* \in T_{v_i}$. Then, $v_i \in R^*$ for some $R^* \in \mathcal{R}_{v^*}$ and $R^* = \bigcap_{v \leq v^*} S_v^{v_i}$.

Proof. Since $v^* \in T_{v_i}$, we have $v^* \in B_{v_i}$ and hence $v^* \in S_v^{v_i}$ for all $v < v_i$. Consequently, for all $v < v^*$, we know that $S_v^{v^*} = S_v^{v_i}$, which implies that $v_i \in B_{v^*} = \bigcap_{v<v^*} S_v^{v^*} = \bigcap_{v<v^*} S_v^{v_i}$. Since $v_i \in B_{v^*}$, it must also be an element of one of the sets in $\mathcal{R}_{v^*} = \mathcal{S}_{v^*} \cap B_{v^*}$; call this set R^*. Since v_i can only be an element of the set $S_{v^*}^{v_i} \in \mathcal{S}_{v^*}$, we then get

$$R^* = B_{v^*} \cap S_{v^*}^{v_i} = \left(\bigcap_{v<v^*} S_v^{v_i} \right) \cap S_{v^*}^{v_i} = \bigcap_{v \leq v^*} S_v^{v_i}. \qquad \blacksquare$$

Using the preceding results, it is now possible to prove Proposition 2.2 and Theorem 2.2.

Proposition 2.2 (Characterization of Boundary and Reduced-Order Sets).
Let $(v_1, \ldots, v_m)$ be an ordering of the non-leaf vertices of a graph $\mathcal{G}_{\preceq} = (V, E)$.

(1) For any $i = 2, \ldots, m$, the set B_{v_i} defined in (2.22) is equal to some set $R \in \mathcal{R}_v$ with $v < v_i$. Consequently, $\mathcal{R}_{v_i} = \mathcal{S}_{v_i} \cap B_{v_i} = \mathcal{S}_{v_i} \cap R$ is a partitioning of the set R.

(2) For any $v_i \in (v_2, \ldots, v_m)$, suppose $T_{v_i} = \{t_1, \ldots, t_n\}$. The vertices V may be written as the union of $n+1$ disjoint sets $A_1, \ldots, A_n$, and B_{v_i}, where the subgraph induced by $A_j \cup \{t_j\}$ is separated from the rest of the graph by vertex t_j.

Proof.

(1) By Lemma A.2, the set T_{v_i} is non-empty. Take any $v \in T_{v_i}$, and according to Lemma A.3 there must exist a set $R \in \mathcal{R}_v$ that contains v_i. Take R^* to be the smallest set (with respect to inclusion) amongst all such sets R. Call v^* the vertex for which $v_i \in R^* \in \mathcal{R}_{v^*}$. We will show that $B_{v_i} = R^*$.

By Lemma A.3, $R^* = \bigcap_{v \leq v^*} S_v^{v_i}$, and recall that $B_{v_i} = \bigcap_{v \in T_{v_i}} S_v^{v_i} = \bigcap_{v<v_i} S_v^{v_i}$. If there is no vertex $v' \in T_{v_i}$ such that $v^* < v' < v_i$, then we are done since $R^* = B_{v_i}$. Suppose there is such a vertex v'. We now show that if there is such a vertex v' then there exists a set $R' \in \mathcal{R}_{v'}$ which contains v_i and such that $R' \subsetneq R^*$. Since this contradicts the minimality of R^*, we can conclude that there is no such v', in which case $R^* = B_{v_i}$ must be true.

By Lemma A.3, since $v' \in T_{v_i}$, the set $R' \in \mathcal{R}_{v'}$ containing v_i can be written as $R' = \bigcap_{v \leq v'} S_v^{v_i}$. In addition, since $v' \in T_{v_i}$ and hence $v' \in B_{v_i}$, we know that $v' \in S_v^{v_i} \cap S_{v'}^{v_i}$ for all $v < v_i$. Then, by Lemma A.1, $S_v^{v_i} \not\subseteq S_{v'}^{v_i}$ for all $v < v_i$, $v \neq v'$, thereby implying that $S_v^{v_i} \cap S_{v'}^{v_i} \subsetneq S_v^{v_i}$. Using this fact and the fact that $v^* < v' < v_i$ gives the following set relationships,

$$\begin{aligned} R' = \bigcap_{v \leq v'} S_v^{v_i} &= \left(\bigcap_{v<v'} S_v^{v_i} \right) \cap S_{v'}^{v_i} \subseteq \left(\bigcap_{v \leq v^*} S_v^{v_i} \right) \cap S_{v'}^{v_i} \\ &= \bigcap_{v \leq v^*} \left(S_v^{v_i} \cap S_{v'}^{v_i} \right) \subsetneq \bigcap_{v \leq v^*} S_v^{v_i} = R^*. \end{aligned}$$

Thus, we have $v_i \in R' \subsetneq R^*$, which is the desired contradiction.

(2) Using Corollary A.1 and one of DeMorgan's laws gives the following,

$$V - B_{v_i} = V - \bigcap_{t \in T_{v_i}} S^{v_i}_t = \bigcup_{t \in T_{v_i}} (V - S^{v_i}_t). \tag{A.9}$$

Defining $A_j \triangleq V - S^{v_i}_{t_j}$ for $t_j \in T_{v_i}$, $j = 1, \ldots, n$, provides the required decomposition for V, *i.e.* $V = A_1 \cup \ldots \cup A_n \cup B_{v_i}$.

Recall that $S^{v_i}_t$ can only be one of two types. Suppose $S^{v_i}_{t_j}$ is Type 1, then

$$\begin{aligned} S^{v_i}_{t_j} &= \bar{S}_u, \text{ where } u \in \chi(t_j) \text{ and } v_i \succeq u \\ \Longrightarrow A_j &= V - \bar{S}_u = S^c_u - \{t_j\}, \end{aligned}$$

and suppose $S^{v_i}_{t_k}$ is Type 2, then

$$\begin{aligned} S^{v_i}_{t_k} &= S^c_{t_k} \cup \{t_k\} \\ \Longrightarrow A_k &= V - \left(S^c_{t_k} \cup \{t_k\}\right) = S_{t_k} - \{t_k\}. \end{aligned}$$

This then implies that the subgraph induced by $A_j \cup \{t_j\} = S^c_u$ is separated from the rest of the graph by vertex t_j, and similarly, the subgraph induced by $A_k \cup \{t_k\} = S_{t_k}$ is separated from the rest of the graph by vertex t_k.

We now show that A_1,. . .,A_n, and B_{v_i} are disjoint. First, note that $B_{v_i} \subset S^{v_i}_{t_j}$ for $j = 1, \ldots, n$, and consequently $A_j \cap B_{v_i} = \emptyset$. Now consider two sets $A_j = V - S^{v_i}_{t_j}$ and $A_k = V - S^{v_i}_{t_k}$. Using Lemma A.2, both $S^{v_i}_{t_j}$ and $S^{v_i}_{t_k}$ cannot be Type 1, and so, we first consider the case where $S^{v_i}_{t_j}$ is Type 1 and $S^{v_i}_{t_k}$ is Type 2. According to Lemma A.2, if $S^{v_i}_{t_j}$ is Type 1 then $t_k \in S^{v_i}_{t_j}$, implying that $A_k = S_{t_k} - \{t_k\} \subset S^{v_i}_{t_j}$. As a result, we get $A_j \cap A_k = (V - S^{v_i}_{t_j}) \cap A_k = \emptyset$ as desired.

The only other case to consider is when both $S^{v_i}_{t_j}$ and $S^{v_i}_{t_k}$ are Type 2, so that

$$\begin{aligned} A_j &= S_{t_j} - \{t_j\} \\ A_k &= S_{t_k} - \{t_k\}. \end{aligned}$$

Again using Lemma A.2, we know that t_j and t_k cannot be comparable, and consequently, $A_j \cap A_k = \emptyset$. ■

Theorem 2.2 (The Reduced-Order Global Markov Property).
Random vectors $\{X_v\}$ satisfy the global Markov property if and only if they satisfy the reduced-order global Markov property.

Proof. If $\{X_v\}$ satisfies the global Markov property then it also satisfies the reduced-order global Markov property because each $R \in \mathcal{R}_{v_i}$ satisfies $R \subset S$ for some $S \in \mathcal{S}_{v_i}$.

We prove the converse by induction. First, $\mathcal{R}_{v_1} = \mathcal{S}_{v_1}$ so the global Markov property holds at vertex v_1. Suppose now that the global Markov property holds at all vertices $v_1, \ldots, v_{i-1}$. We show that this fact in conjunction with the requirement $\perp X_{\mathcal{R}_{v_i}}$ gives $\perp X_{\mathcal{S}_{v_i}}$. Recall that $\cap \mathcal{S}_{v_i} = \{v_i\}$, and suppose $\mathcal{S}_{v_i} - \cap\mathcal{S}_{v_i} = \mathcal{S}_{v_i} - \{v_i\} = \{S_1, \ldots, S_k\}$. Consequently, in order for $\perp X_{\mathcal{S}_{v_i}}$ to hold, we need

$$p(x_{S_1}, \ldots, x_{S_k} | x_{v_i}) = \prod_{j=1}^{k} p(x_{S_j} | x_{v_i}).$$

Consider the reduced-order sets $\mathcal{R}_{v_i}$, and suppose $\mathcal{R}_{v_i} - \cap\mathcal{R}_{v_i} = \mathcal{R}_{v_i} - \{v_i\} = \{R_1, \ldots, R_k\}$, where $R_j \subset S_j$ for $j = 1, \ldots, k$. Further, define $U_j \triangleq S_j - R_j$, for $j = 1, \ldots, k$. Then,

$$\begin{aligned} p(x_{S_1}, \ldots, x_{S_k} | x_{v_i}) &= p(x_{R_1}, \ldots, x_{R_k}, x_{U_1}, \ldots, x_{U_k} | x_{v_i}) \\ &= p(x_{R_1}, \ldots, x_{R_k} | x_{v_i}) p(x_{U_1}, \ldots, x_{U_k} | x_{v_i}, x_{R_1}, \ldots, x_{R_k}) \\ &= \left[\prod_{j=1}^{k} p(x_{R_j} | x_{v_i}) \right] p(x_{U_1}, \ldots, x_{U_k} | x_{v_i}, x_{R_1}, \ldots, x_{R_k}), \end{aligned} \tag{A.10}$$

where the last equality is due to the reduced-order global Markov property $\perp X_{\mathcal{R}_{v_i}}$ holding at vertex v_i. Using the chain rule for probabilities, the last term in (A.10) may be written as follows,

$$p(x_{U_1}, \ldots, x_{U_k} | x_{v_i}, x_{R_1}, \ldots, x_{R_k}) = \prod_{j=1}^{k} p(x_{U_j} | x_{U_1}, \ldots, x_{U_{j-1}}, x_{v_i}, x_{R_1}, \ldots, x_{R_k}). \tag{A.11}$$

Choose any term $p(x_{U_j} | x_{U_1}, \ldots, x_{U_{j-1}}, x_{v_i}, x_{R_1}, \ldots, x_{R_k})$ in (A.11); we will show that this term is equal to $p(x_{U_j} | x_{v_i}, x_{R_j})$.

To see this, suppose $T_{v_i} = \{t_1, \ldots, t_n\}$, and note the following about the set U_j,

$$\begin{aligned} U_j &= S_j - R_j = S_j - S_j \cap B_{v_i} = S_j \cap (V - B_{v_i}) \\ &= S_j \cap (A_1 \cup \cdots \cup A_n), \end{aligned} \tag{A.12}$$

where the sets $A_1, \ldots, A_n$ in the final equality are the same as those considered in the second part of Proposition 2.2. We now show that each set A_l, $l = 1, \ldots, n$, must either be completely contained in the set S_j, or not contained at all, *i.e* the intersection $A_l \cap S_j$ is either A_l or $\emptyset$.

Recall from the proof of Proposition 2.2 that A_l can have one of two forms. First, suppose $A_l = S_{t_l} - \{t_l\}$, and recall that A_l and B_{v_i} must be disjoint. Consequently, A_l cannot contain vertex v_i, implying that $v_i \not\succ t_l$. This in turn implies that A_l must be completely contained in one of the sets $S \in \mathcal{S}_{v_i} - \{v_i\}$. Thus, either A_l is a subset of S_j, or $A_l \cap S_j = \emptyset$ as desired. Finally, suppose $A_l = S_u^c - \{t_l\}$ is of the second form described in Proposition 2.2, where $u \in \chi(t_l)$ and $v_i \succeq u$. This immediately implies that A_l is a subset of the Type 2 set contained in the family $\mathcal{S}_{v_i}$. Thus, $A_l \cap S_j = \emptyset$ if $S_j \cup \{v_i\}$ is Type 1, and $A_l \subset S_j$ if $S_j \cup \{v_i\}$ is Type 2.

Using (A.12) along with the preceding discussion, indicates that U_j is the union of the sets A_l, $l = 1, \ldots, n$, which are completely contained in S_j. Suppose $A_l \subset S_j$ for $l = l_1, \ldots, l_p$, so that $U_j = A_{l_1} \cup \cdots \cup A_{l_p}$. From Proposition 2.2, we also know that each vertex t_l, $l = 1, \ldots, n$, separates the subgraph induced by $A_l \cup \{t_l\}$ from the rest of the graph. Furthermore, since $t_l \in T_{v_i}$, we know that $t_l < v_i$, and by our assumption, the global Markov property must then hold at each of the vertex t_l, $l = 1, \ldots, n$.

All of this implies that $p(x_{U_j}|x_{V-U_j}) = p(x_{U_j}|x_{t_{l_1}}, x_{t_{l_2}}, \ldots, x_{t_{l_p}})$, and since $t_{l_1}, \ldots, t_{l_p} \in R_j$, we can also write $p(x_{U_j}|x_{V-U_j}) = p(x_{U_j}|x_{R_j}, x_{v_i})$. Using this fact in (A.10) and (A.11) gives the result,

$$\begin{aligned} p(x_{S_1}, \ldots, x_{S_k}|x_{v_i}) &= \left[\prod_{j=1}^{k} p(x_{R_j}|x_{v_i})\right] \left[\prod_{j=1}^{k} p(x_{U_j}|x_{v_i}, x_{R_j})\right] \\ &= \prod_{j=1}^{k} p(x_{R_j}, x_{U_j}|x_{v_i}) = \prod_{j=1}^{k} p(x_{S_j}|x_{v_i}). \end{aligned}$$ ■

■ A.3 Proof of Propositions 2.3 and 2.4

Proposition 2.3 (Marginalization-Invariant Markovianity and a Top-Down Ordering). *Suppose the marginalization constraint set M is equal to all leaf vertices of a graph $\mathcal{G}_{\preceq}$, and let $(v_1, \ldots, v_m)$ be a top-down ordering of the non-leaf vertices. Then, the families $\mathcal{M}_{v_i}$ may be written as follows:*

$$\begin{aligned} \mathcal{M}_{v_0} - \{v_0\} &= \{L_v\}_{v\in\chi(v_0)}, \\ \mathcal{M}_{v_i} - \{v_i\} &= \{L_v\}_{v\in\chi(v_i)} \cup \{\pi(v_i)\}, \quad v_i \neq v_0. \end{aligned}$$

Proof. Since the ordering is top-down, v_0 must appear first in the ordering, which implies that $\mathcal{R}_{v_0} = \mathcal{S}_{v_0} = \{\bar{S}_v\}_{v\in\chi(v_0)}$. Since M contains all leaf vertices, we have $M^{(1)} = M \cup \{v_0\} = L_{v_0} \cup \{v_0\}$, implying the following,

$$\begin{aligned} \mathcal{M}_{v_0} &= \mathcal{R}_{v_0} \cap M^{(1)} = \{\bar{S}_v\}_{v\in\chi(v_0)} \cap (L_{v_0} \cup \{v_0\}) \\ &= \{\bar{S}_v \cap (L_{v_0} \cup \{v_0\}\}_{v\in\chi(v_0)} = \{L_v \cup \{v_0\}\}_{v\in\chi(v_0)}. \end{aligned}$$

Consider now any other vertex v_i in the ordering, and recall that $B_{v_i} = \bigcap_{v<v_i} S_v^{v_i}$. Since the ordering is top-down, $\pi(v_i)$ must appear before v_i in the ordering, and the set $S^{v_i}_{\pi(v_i)}$ is equal to $\bar{S}_{v_i}$. Consequently, $B_{v_i} \subset \bar{S}_{v_i}$. However, due to the top-down ordering, there is no $v_j \in \bar{S}_{v_i}$ with $v_j < v_i$, which then implies that $B_{v_i} = \bar{S}_{v_i}$. This then gives,

$$\mathcal{R}_{v_i} = \mathcal{S}_{v_i} \cap \bar{S}_{v_i} = \{\bar{S}_v\}_{v\in\chi(v_i)} \cup \{\{v_i\} \cup \pi(v_i)\}.$$

By definition, $M^{(i)}$ contains all leaf vertices plus all vertices less than v_i, but since $\cup\mathcal{R}_{v_i} = \bar{S}_{v_i}$, the only elements of $M^{(i)}$ contained in $\cup\mathcal{R}_{v_i}$ are L_{v_i}, $\pi(v_i)$, and v_i. This gives the needed characterization as follows,

$$\begin{aligned} \mathcal{M}_{v_i} &= \mathcal{R}_{v_i} \cap M^{(i)} = \{\bar{S}_v \cap M^{(i)}\}_{v\in\chi(v_i)} \cup \{(\{v_i\} \cup \pi(v_i)) \cap M^{(i)}\} \\ &= \{L_v \cup \{v_i\}\}_{v\in\chi(v_i)} \cup \{\{v_i\} \cup \pi(v_i)\}. \end{aligned}$$ ■

We use the following lemma, in order to later prove Proposition 2.4. We now choose to define the boundary set $B_{v_1} \triangleq V$ for the first vertex v_1 in any ordering $(v_1, \ldots, v_m)$.

Lemma A.4 (Useful Results for Bottom-up Ordering).
Suppose the marginalization constraint set M is equal to all leaf vertices of a graph $\mathcal{G}_{\preceq}$, and let $(v_1, \ldots, v_m)$ be a bottom-up ordering of the non-leaf vertices. Then, the following two equalities hold for $1 \leq i \leq m$,

(1) $\min\limits_{\mathcal{G}_{\preceq}}\left(M^{(i)}\right) = \left[S^c_{v_i} \cap \min\limits_{\mathcal{G}_{\preceq}}\left(M^{(i-1)}\right)\right] \cup \{v_i\}$,

(2) $B_{v_i} \cap M^{(i)} = \min\limits_{\mathcal{G}_{\preceq}}\left(M^{(i-1)}\right) \cup \{v_i\}$.

Proof.

(1) By the nature of the bottom-up ordering, $v_i \in \min_{\mathcal{G}_{\preceq}}\left(M^{(i)}\right)$ since there is no element smaller than v_i in $M^{(i)}$. For all $v_j \succ v_i$, $v_j \notin \min_{\mathcal{G}_{\preceq}}\left(M^{(i)}\right)$ by definition, and consequently, we may safely remove all descendants of v_i from $M^{(i)}$,

$$\begin{aligned}\min_{\mathcal{G}_{\preceq}}\left(M^{(i)}\right) &= \min_{\mathcal{G}_{\preceq}}\left(M^{(i-1)} \cup \{v_i\}\right) = \min_{\mathcal{G}_{\preceq}}\left(\left(S^c_{v_i} \cap M^{(i-1)}\right) \cup \{v_i\}\right)\\ &= \min_{\mathcal{G}_{\preceq}}\left(S^c_{v_i} \cap M^{(i-1)}\right) \cup \{v_i\} = \left[S^c_{v_i} \cap \min_{\mathcal{G}_{\preceq}}\left(M^{(i-1)}\right)\right] \cup \{v_i\}.\end{aligned}$$

(2) Note that $B_{v_1} \cap M^{(1)} = M^{(1)} = M^{(0)} \cup \{v_1\} = \min_{\mathcal{G}_{\preceq}}\left(M^{(0)}\right) \cup \{v_1\}$, where the last equality is due to the fact that $M^{(0)} = M$ only contains leaf vertices. Suppose now that $B_{v_{i-1}} \cap M^{(i-1)} = \min_{\mathcal{G}_{\preceq}}\left(M^{(i-2)}\right) \cup \{v_{i-1}\}$, and consider the following decomposition of $B_{v_i} \cap M^{(i)}$,

$$B_{v_i} \cap M^{(i)} = \left(\bigcap_{v<v_i} S^{v_i}_v\right) \cap \left(M^{(i-1)} \cup \{v_i\}\right) = \left[\left(\bigcap_{v<v_i} S^{v_i}_v\right) \cap M^{(i-1)}\right] \cup \{v_i\}.$$

Due to the bottom-up ordering, for any $v < v_i$, either $v_i \prec v$, or v_i and v are not comparable, and as such, we must have $S^{v_i}_v = S^c_v \cup \{v\}$, for all $v < v_i$. Using this fact, gives the following,

$$\begin{aligned}B_{v_i} \cap M^{(i)} &= \left[\left(\bigcap_{v<v_i} \left(S^c_v \cup \{v\}\right)\right) \cap M^{(i-1)}\right] \cup \{v_i\}\\ &= \left[\left(S^c_{v_{i-1}} \cup \{v_{i-1}\}\right) \cap \left(\bigcap_{v<v_{i-1}} \left(S^c_v \cup \{v\}\right)\right) \cap M^{(i-1)}\right] \cup \{v_i\}\\ &= \left[\left(S^c_{v_{i-1}} \cup \{v_{i-1}\}\right) \cap B_{v_{i-1}} \cap M^{(i-1)}\right] \cup \{v_i\}.\end{aligned}$$

Using the induction hypothesis and part (1) of this lemma implies the result,

$$\begin{aligned}B_{v_i} \cap M^{(i)} &= \left[\left(S^c_{v_{i-1}} \cup \{v_{i-1}\}\right) \cap \left(\min_{\mathcal{G}_{\preceq}}\left(M^{(i-2)}\right) \cup \{v_{i-1}\}\right)\right] \cup \{v_i\}\\ &= \left[\left(S^c_{v_{i-1}} \cap \min_{\mathcal{G}_{\preceq}}\left(M^{(i-2)}\right)\right) \cup \{v_{i-1}\}\right] \cup \{v_i\} = \min_{\mathcal{G}_{\preceq}}\left(M^{(i-1)}\right) \cup \{v_i\}.\end{aligned}$$ ■

Proposition 2.4 (Marginalization-Invariant Markovianity and a Bottom-Up Ordering). *Suppose the marginalization constraint set M is equal to all leaf vertices of a graph $\mathcal{G}_{\preceq}$, and let $(v_1, \ldots, v_m)$ be a bottom-up ordering of the non-leaf vertices. Then, the families $\mathcal{M}_{v_i}$ may be written as follows:*

$$\begin{aligned}\mathcal{M}_{v_0} &= \{\{v_0, v\}\}_{v\in\chi(v_0)},\\ \mathcal{M}_{v_i} &= \{\{v_i, v\}\}_{v\in\chi(v_i)} \cup \left\{\min_{\mathcal{G}_{\preceq}}\left(M^{(i)}\right)\right\}, \quad v_i \neq v_0.\end{aligned}$$

Proof. Notice that $\mathcal{M}_{v_i} = \mathcal{R}_{v_i} \cap M^{(i)} = (\mathcal{S}_{v_i} \cap B_{v_i}) \cap M^{(i)}$ for all $1 \leq i \leq m$. The second part of Lemma A.4 implies the following,

$$\mathcal{M}_{v_i} = \mathcal{S}_{v_i} \cap \left(B_{v_i} \cap M^{(i)}\right) = \mathcal{S}_{v_i} \cap \left(\min_{\mathcal{G}_{\preceq}}\left(M^{(i-1)}\right) \cup \{v_i\}\right). \tag{A.15}$$

Consider in turn the intersection of each set in $\mathcal{S}_{v_i}$ with $\min_{\mathcal{G}_{\preceq}}\left(M^{(i-1)}\right) \cup \{v_i\}$. First, for any vertex $v \in \chi(v_i)$, we must characterize the intersection

$$\begin{aligned}\bar{S}_v \cap \left(\min_{\mathcal{G}_{\preceq}}\left(M^{(i-1)}\right) \cup \{v_i\}\right) &= (S_v \cup \{v_i\}) \cap \left(\min_{\mathcal{G}_{\preceq}}\left(M^{(i-1)}\right) \cup \{v_i\}\right) \\ &= \left(S_v \cap \min_{\mathcal{G}_{\preceq}}\left(M^{(i-1)}\right)\right) \cup \{v_i\},\end{aligned}$$

but since the ordering is bottom-up, the only element of $\min_{\mathcal{G}_{\preceq}}\left(M^{(i-1)}\right)$ also contained in S_v is v. This provides the first part of the characterization of $\mathcal{M}_{v_i}$. Second, if v_i is not the root vertex then we must also characterize the intersection

$$(S_{v_i}^c \cup \{v_i\}) \cap \left(\min_{\mathcal{G}_{\preceq}}\left(M^{(i-1)}\right) \cup \{v_i\}\right) = \left(S_{v_i}^c \cap \min_{\mathcal{G}_{\preceq}}\left(M^{(i-1)}\right)\right) \cup \{v_i\},$$

but by the first part of Lemma A.4, this equals $\min_{\mathcal{G}_{\preceq}}\left(M^{(i)}\right)$. ∎

■ A.4 Proof of Proposition 2.6 and Proposition 2.8

Proposition 2.6 (Nested Marginalization-Invariant Constraints).
Let $\mathcal{G}_{\preceq}$ be a rooted tree, and let $(v_1, \ldots, v_m)$ be an arbitrary ordering on the non-leaf vertices of $\mathcal{G}_{\preceq}$. Given any marginalization constraint set $M \subseteq V$, the sets $\cup\mathcal{M}_{v_i}$ are nested in the following sense:

$$\cup\mathcal{M}_{v_1} - \{v_1\} \subseteq M \tag{A.16a}$$

$$\cup\mathcal{M}_{v_i} - \{v_i\} \subseteq \cup\mathcal{M}_{v_j} \text{ for some } v_j < v_i, \text{ and } i = 2, \ldots, m. \tag{A.16b}$$

Proof. First, note that $\mathcal{M}_{v_1} = \mathcal{R}_{v_1} \cap M^{(1)} = \mathcal{S}_{v_1} \cap M^{(1)}$. Consequently, $\cup\mathcal{M}_{v_1} = M^{(1)} = M \cup \{v_1\}$. If $v_1 \notin M$ then $\cup\mathcal{M}_{v_1} - \{v_1\} = M$, otherwise $\cup\mathcal{M}_{v_1} - \{v_1\} \subset M$.

In a similar manner, note that $\mathcal{M}_{v_i} = \mathcal{R}_{v_i} \cap M^{(i)} = \mathcal{S}_{v_i} \cap B_{v_i} \cap M^{(i)}$, and consequently, $\cup\mathcal{M}_{v_i} = B_{v_i} \cap M^{(i)}$ and $\cup\mathcal{M}_{v_i} - \{v_i\} \subseteq B_{v_i} \cap M^{(i-1)}$ (with equality if $v_i \notin M$). Using the first part of Proposition 2.2, we know that $B_{v_i} = R$ for some $R \in \mathcal{R}_{v_j}$ and $v_j < v_i$. Furthermore, since $v_i \in R$ and $R \in \mathcal{R}_{v_j}$, we can equivalently write $B_{v_i} = S_{v_j}^{v_i} \cap B_{v_j}$. Using a result in the proof of Proposition 2.2, we can also write $B_{v_i} = \cap_{v \leq v_j} S_v^{v_i}$, and using Corollary A.1, we can write $B_{v_i} = \cap_{v \in T_{v_i}} S_v^{v_i}$. In summary, we have the following forms for B_{v_i},

$$B_{v_i} = \bigcap_{v < v_i} S_v^{v_i} = \bigcap_{v \leq v_j} S_v^{v_i} = \bigcap_{v \in T_{v_i}} S_v^{v_i} = S_{v_j}^{v_i} \cap B_{v_j}. \tag{A.17}$$

Using the first three equalities in (A.17), we see that the vertices $v_{j+1}, \ldots, v_{i-1}$ are not elements of T_{v_i}, and consequently, $v_{j+1}, \ldots, v_{i-1} \notin B_{v_i}$. Using this fact and the fourth equality in (A.17) provides the result for $i = 2, \ldots, m$,

$$\begin{aligned}\cup\mathcal{M}_{v_i} - \{v_i\} &\subseteq B_{v_i} \cap M^{(i-1)} = B_{v_i} \cap \left(M^{(j)} \cup \{v_{j+1}, \ldots, v_{i-1}\}\right) \\ &= B_{v_i} \cap M^{(j)} = S_{v_j}^{v_i} \cap B_{v_j} \cap M^{(j)} = S_{v_j}^{v_i} \cap \left(\cup\mathcal{M}_{v_j}\right) \subseteq \cup\mathcal{M}_{v_j}.\end{aligned}$$ ∎

Proposition 2.8 (Properties of Augmented Marginalization-Invariant Families).
Let $\mathcal{G}_{\preceq}$ be a rooted tree, and let $(v_1, \ldots, v_m)$ be an arbitrary ordering on the non-leaf vertices of $\mathcal{G}_{\preceq}$. Given *any* *marginalization constraint set $M \subseteq V$, the following is true:*

(1) The constraints $\perp X_{\mathcal{M}^\sharp_{v_1}}, \ldots, \perp X_{\mathcal{M}^\sharp_{v_m}}$ are ordered with respect to $M^\sharp$.

(2) The sets $\cup\mathcal{M}^\sharp_{v_i}$ are nested in the following sense:

$$\cup\mathcal{M}^\sharp_{v_1} - \{v_1^{(d)}\} = M^\sharp \tag{A.18a}$$

$$\cup\mathcal{M}^\sharp_{v_i} - \{v_i^{(d)}\} \subseteq \cup\mathcal{M}^\sharp_{v_j} \textit{ for some } v_j < v_i, \textit{ and } i = 2, \ldots, m. \tag{A.18b}$$

Proof.

(1) Note that either $\cap\mathcal{M}^\sharp_{v_i} = \{v_i^{(d)}, v_i^{(t)}\}$ or $\cap\mathcal{M}^\sharp_{v_i} = \{v_i^{(d)}\}$ depending on whether or not $v_i \in M$. By definition, $M^\sharp$ does not contain any design vertices, and therefore, $v_i^{(d)} \notin M^\sharp$. Similarly, by definition, $v_i^{(d)} \notin \cup\mathcal{M}^\sharp_{v_j}$ for all $v_j < v_i$.

(2) Using the proof to Proposition 2.6, we know that $\cup\mathcal{M}_{v_1} = M \cup \{v_1\}$. Suppose we apply the same steps used in the augmentation rule (for the family $\mathcal{M}_{v_1}$) to the set $\cup\mathcal{M}_{v_1}$, meaning the following for each $v \in \cup\mathcal{M}_{v_1}$:

   (a) If $v \neq v_1$, replace v with $v^{(t)}$.

   (b) If $v_1 \in M$, replace v_1 with the tuple $v_1^{(d)}, v_1^{(t)}$; otherwise, replace v_1 with $v_1^{(d)}$.

   By definition, these steps generate the set $\cup\mathcal{M}^\sharp_{v_1}$. Consider now applying operations (a) and (b) to each $v \in M \cup \{v_1\}$. If $v_1 \in M$, these operations generate the set $M^\sharp \cup \{v_1^{(d)}\}$, and if $v_1 \notin M$, these operations also generate the set $M^\sharp \cup \{v_1^{(d)}\}$. Consequently, we get $\cup\mathcal{M}^\sharp_{v_1} = M^\sharp \cup \{v_1^{(d)}\}$, and since $v_1^{(d)} \notin M^\sharp$, this is equivalent to $\cup\mathcal{M}^\sharp_{v_1} - \{v_1^{(d)}\} = M^\sharp$, thereby validating (A.18a).

   Next, using Proposition 2.6, we know that $\cup\mathcal{M}_{v_i} - \{v_i\} \subseteq \cup\mathcal{M}_{v_j}$ for some $v_j < v_i$ and $i = 2, \ldots, m$, and furthermore, from the proof to the proposition, we know that the vertices $v_{j+1}, \ldots, v_i$ are not contained in $\cup\mathcal{M}_{v_i} - \{v_i\}$. Suppose we apply the same steps used in the augmentation rule (for the family $\mathcal{M}_{v_j}$) to the set $\cup\mathcal{M}_{v_j}$, meaning the following for each $v \in \cup\mathcal{M}_{v_j}$:

   (a) If v is a leaf vertex or if v is a non-leaf vertex with $v > v_j$, replace v with $v^{(t)}$.

   (b) If v is a non-leaf vertex with $v \leq v_j$ and $v \in M$, replace v with the tuple $v^{(d)}, v^{(t)}$.

   (c) If v is a non-leaf vertex with $v \leq v_j$ and $v \notin M$, replace v with $v^{(d)}$.

   By definition, these steps generate the set $\cup\mathcal{M}^\sharp_{v_j}$. Consider now applying operations (a), (b), and (c) to each $v \in \cup\mathcal{M}_{v_i} - \{v_i\}$. Since vertices $v_{j+1}, \ldots, v_i$ are not contained in $\cup\mathcal{M}_{v_i} - \{v_i\}$, operations (a), (b), and (c) are equivalent to the following operations for each $v \in \cup\mathcal{M}_{v_i} - \{v_i\}$:

   (a') If v is a leaf vertex or if v is a non-leaf vertex with $v > v_i$, replace v with $v^{(t)}$.

   (b') If v is a non-leaf vertex with $v \leq v_i$ and $v \in M$, replace v with the tuple $v^{(d)}, v^{(t)}$.

(c') If v is a non-leaf vertex with $v \leq v_i$ and $v \notin M$, replace v with $v^{(d)}$.

If $v_i \notin M$, these steps generate the set $\cup\mathcal{M}^{\sharp}_{v_i} - \{v_i^{(d)}\}$, thereby validating (A.18b).

If $v_i \in M$, these steps generate the set $\cup\mathcal{M}^{\sharp}_{v_i} - \{v_i^{(d)}, v_i^{(t)}\}$. However, since $v_i \in M$, we have $v_i \in M^{(j)}$, and using (A.17), we also have $v_i \in B_{v_j}$. This implies that $v_i \in \cup\mathcal{M}_{v_j} = B_{v_j} \cap M^{(j)}$. Finally, since $v_i \in \cup\mathcal{M}_{v_j}$, the augmentation rule implies that $v_i^{(t)} \in \cup\mathcal{M}^{\sharp}_{v_j}$. Consequently, we get $\cup\mathcal{M}^{\sharp}_{v_i} - \{v_i^{(d)}, v_i^{(t)}\} \subset \cup\mathcal{M}^{\sharp}_{v_i} - \{v_i^{(d)}\} \subseteq \cup\mathcal{M}^{\sharp}_{v_j}$, thereby validating (A.18b) for the case $v_i \in M$. ■

Appendix B

Proofs for Chapter 3

B.1 Proof of Proposition 3.5

Proposition 3.5 (Vertex Elimination and Junction Trees).
Let $\mathcal{G} = (V, E)$ be a triangulated graph, and let $\mathcal{T} = (\mathcal{C}, \mathcal{S})$ be any junction tree representation of $\mathcal{G}$. Suppose $v \in V$ is a simplicial vertex, and let C denote the unique maximal clique containing v. If we define the elimination graph $\mathcal{G}^{\downarrow} \triangleq \downarrow(\mathcal{G}, v)$ as well as the sets $C^{\downarrow} \triangleq C - \{v\}$ and $\mathcal{C}^{\downarrow} \triangleq \mathcal{C} - \{C\}$, then one and only one of the following is a junction tree representation $\mathcal{T}^{\downarrow}$ for $\mathcal{G}^{\downarrow}$:

(1) $\mathcal{T}^{\downarrow} = (\mathcal{C}^{\downarrow} \cup \{C^{\downarrow}\}, \mathcal{S})$,

(2) $\mathcal{T}^{\downarrow} = (\mathcal{C}^{\downarrow}, \mathcal{S} - \{C^{\downarrow}\})$.

Proof. First, by the second part of Lemma 3.1, we know that $C = N_{\mathcal{G}}[v]$ is the unique maximal clique in $\mathcal{G}$ containing v. If we eliminate v from $\mathcal{G}$, then either $C - \{v\}$ is a maximal clique of $\mathcal{G}^{\downarrow}$, or it is a subset of another maximal clique. First, suppose that $C - \{v\}$ is a maximal clique of $\mathcal{G}^{\downarrow}$. Then, the set of maximal cliques of $\mathcal{G}^{\downarrow}$ consists of all maximal cliques in $\mathcal{G}$ except C, plus the new maximal clique $C - \{v\}$, *i.e.* $(\mathcal{C} - \{C\}) \cup \{C - \{v\}\} = \mathcal{C}^{\downarrow} \cup \{C^{\downarrow}\}$. Next, since v is only an element of the maximal clique C, it is not an element of any separator set $S \in \mathcal{S}$. Since $\mathcal{T}$ satisfies the running intersection property then $\mathcal{T}^{\downarrow} = (\mathcal{C}^{\downarrow} \cup \{C^{\downarrow}\}, \mathcal{S})$ satisfies this property as well, proving that this is the junction tree representation for $\mathcal{G}^{\downarrow}$.

Now, suppose that $C - \{v\}$ is a subset of at least one other maximal clique in $\mathcal{G}^{\downarrow}$, and suppose we are given any junction tree for the graph $\mathcal{G}$. In such a junction tree, C has $n \geq 1$ neighbors $\{C_1, \ldots, C_n\}$ and corresponding separator sets $\{S_1, \ldots, S_n\}$, as graphically illustrated in Figure B.1(a). From the running intersection property of the junction tree, we know that C must be connected to at least one of the maximal cliques $\bar{C}$ which satisfy $C - \{v\} \subseteq \bar{C}$. To see this, assume that this is not the case. Then, take any such $\bar{C}$, and note that $C \cap \bar{C} = C - \{v\}$. Consider any path from C to $\bar{C}$, and note that it must pass through one of the maximal cliques C_i. Since no C_i satisfies $C \cap \bar{C} = C - \{v\} \subseteq C_i$, this cannot be a junction tree which is a contradiction. Therefore, we must have $C - \{v\} \subseteq C_i$ for some neighboring vertex C_i in the junction tree; we henceforth assume that C_1 satisfies this requirement.

If C is not a leaf vertex in the junction tree (*i.e.* $n = 1$), we now show that there exists a junction tree such that this is the case, as illustrated in Figure B.1(b). Specifically, we disconnect the cliques $C_2, \ldots, C_n$ from C and reconnect them to C_1, and we call the new separator sets $S'_1, \ldots, S'_n$. In order to prove that this is a junction tree, we will show that the graph in Figure B.1(b) is a tree and that $S_i = S'_i$ for $i = 1, \ldots, n$. Hence, we still have a maximal weight spanning tree and therefore a junction tree.

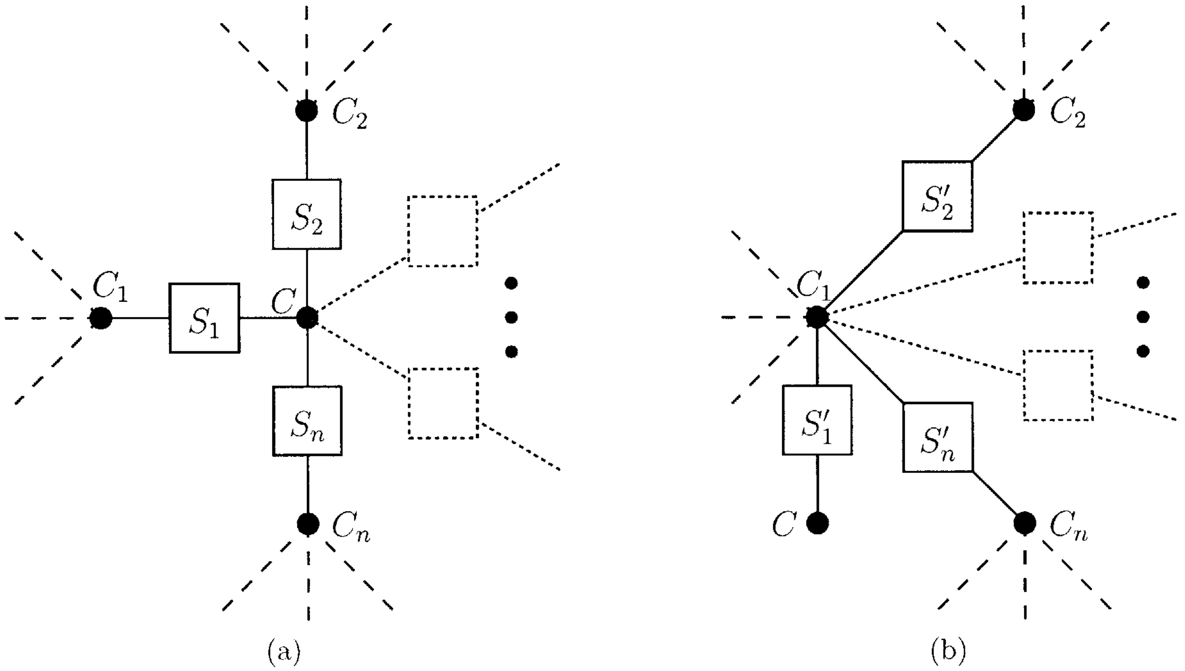


Figure B.1. (a) A junction tree for the triangulated graph $\mathcal{G}$ considered in the proof to Proposition 3.5. Specifically, C is the unique maximal clique containing a vertex v, and C has n neighbors $C_1, \ldots, C_n$ in the junction tree. (b) This graph is obtained from the graph in (a) by disconnecting C_i, $i = 2, \ldots, n$, from C and reconnecting C_i to C_1. Assuming $C - \{v\} \subseteq C_1$, we show that this graph is also a junction tree for $\mathcal{G}$.

First, the graph in Figure B.1(b) must be a tree because it is connected and has the same number of edges as the tree in Figure B.1(a) (see Lemma 3.2). Notice also that $S'_1 = S_1 = C - \{v\}$. Consider now the sets S'_i for $i = 2, \ldots, n$. Since the graph in Figure B.1(a) is a junction tree, the running intersection property indicates that $C_i \cap C_1 \subseteq C$ for $i = 2, \ldots, n$, which implies the following,

$$C_i \cap C_1 = C_i \cap (C_i \cap C_1) \subseteq C_i \cap C. \tag{B.1}$$

In addition, we know that $C - \{v\} \subseteq C_1$, which then implies the following,

$$C_i \cap (C - \{v\}) = C_i \cap C \subseteq C_i \cap C_1. \tag{B.2}$$

The relationships in (B.1) and (B.2) then imply that $S'_i = C_i \cap C_1 = C_i \cap C = S_i$ for $i = 2, \ldots, n$, and therefore, Figure B.1(b) is a junction tree for $\mathcal{G}$.

Given that the junction tree in Figure B.1(b) has C has a leaf vertex, consider now removing vertex v from the graph $\mathcal{G}$. If we do so, then the clique $C - \{v\} = S_1$ is no longer maximal, and this clique as well as the separator S_1 may be removed from the junction tree shown in Figure B.1(b) to give a junction tree for $\mathcal{G}^{\downarrow}$. ■

B.2 Proof of Corollary 3.1

Corollary 3.1 (k-Partial Elimination Orderings and Marginalization).
Let $\mathcal{G} = (V, E)$ be a triangulated graph. Suppose α is a k-partial elimination ordering, and define the elimination graph $\mathcal{G}^{\downarrow} \triangleq \mathcal{G}\left\langle V - \{\alpha(1), \ldots, \alpha(k)\}\right\rangle$. Then, the following decomposition holds for $p_{\mathcal{G}}(x)$,

$$p_{\mathcal{G}}(x) = p_{\mathcal{G}^{\downarrow}}(x) \prod_{i=1}^{k} p\left(x_{\alpha(i)} | x_{N_{\mathcal{G}}^{\downarrow}(\alpha(i))}\right) = p_{\mathcal{G}}\left(x_{V-\{\alpha(1),\ldots,\alpha(k)\}}\right) \prod_{i=1}^{k} p\left(x_{\alpha(i)} | x_{N_{\mathcal{G}}^{\downarrow}(\alpha(i))}\right). \qquad \text{(B.3)}$$

Proof. Given a k-partial elimination ordering α, consider the sequence of elimination graphs $\mathcal{G}_i^{\downarrow}$ given in (3.27). In any elimination graph $\mathcal{G}_{i-1}^{\downarrow}$ with $i \leq k$, we know that $D_{\mathcal{G}_{i-1}^{\downarrow}}(\alpha(i)) = D_{\mathcal{G}}^{\downarrow}(\alpha(i)) = \emptyset$, and therefore, we use (3.36) to give the following,

$$p_{\mathcal{G}_{i-1}^{\downarrow}}(x) = p_{\mathcal{G}_i^{\downarrow}}(x) p(x_{\alpha(i)} | x_{N_{\mathcal{G}_{i-1}^{\downarrow}}(\alpha(i))}) = p_{\mathcal{G}_i^{\downarrow}}(x) p(x_{\alpha(i)} | x_{N_{\mathcal{G}}^{\downarrow}(\alpha(i))}), \quad i = 1, \ldots, k.$$

Applying the above relationship recursively proves the first equality in (B.3),

$$p_{\mathcal{G}}(x) = p_{\mathcal{G}_0^{\downarrow}}(x) = p_{\mathcal{G}_k^{\downarrow}}(x) \prod_{i=1}^{k} p\left(x_{\alpha(i)} | x_{N_{\mathcal{G}}^{\downarrow}(\alpha(i))}\right) = p_{\mathcal{G}^{\downarrow}}(x) \prod_{i=1}^{k} p\left(x_{\alpha(i)} | x_{N_{\mathcal{G}}^{\downarrow}(\alpha(i))}\right). \qquad \text{(B.4)}$$

To prove the second equality in (B.3), we integrate out the variables $x_{\alpha(1)}, \ldots, x_{\alpha(k)}$ (in that order) from both sides of (B.4). Such an integration yields $p_{\mathcal{G}}(x_{V-\{\alpha(1),\ldots,\alpha(k)\}})$ on the left side of (B.4). In integrating the right side of (B.4), note that by definition, variable $x_{\alpha(i)}$ is not included in any terms $p\left(x_{\alpha(j)} | x_{N_{\mathcal{G}}^{\downarrow}(\alpha(j))}\right)$ with $j > i$. Consequently, integrating the right side of (B.4) gives $p_{\mathcal{G}^{\downarrow}}(x)$, thereby proving the second equality in (B.3). ■

B.3 Proof of Proposition 3.7

Proposition 3.7 (Elimination Graphs and Clique Extensions).
Let $\mathcal{G} = (V, E)$ be a triangulated graph. For some $v \in V$, let $F \subseteq D_{\mathcal{G}}(v)$, $F \neq \{\emptyset\}$, and define the new graph $\mathcal{G}' \triangleq (V, E \cup F)$. Then, $\mathcal{G}'$ is a clique extension of $\mathcal{G}$ if and only if there exists a k-partial elimination ordering α of $\mathcal{G}$ such that $F = D_{\mathcal{G}^{\downarrow}}(v)$, with $\mathcal{G}^{\downarrow} \triangleq \mathcal{G}\left\langle V - \{\alpha(1), \ldots, \alpha(k)\}\right\rangle$. Furthermore, the unique new maximal clique C contained in $\mathcal{G}'$ is given by $C = N_{\mathcal{G}^{\downarrow}}[v]$.

Proof. Suppose $\mathcal{G}'$ is a clique extension of $\mathcal{G}$, and let C be the unique new maximal clique contained in $\mathcal{G}'$ but not $\mathcal{G}$. From Lemma 3.4, we know that there exists a perfect elimination ordering α of $\mathcal{G}'$ down to the clique C, and by the second part of Proposition 3.9, α is also a partial elimination ordering for $\mathcal{G}$ down to C. Let $k = |V| - |C|$, and define $\mathcal{G}^{\downarrow} \triangleq \mathcal{G}\left\langle V - \{\alpha(1), \ldots, \alpha(k)\}\right\rangle = \mathcal{G}\langle C\rangle$. Using Proposition 3.8, every edge $\{a, b\} \in F$ satisfies $\{a, b\} \subset C$, and since $F \subseteq D_{\mathcal{G}}(v)$, we must also have $F \subseteq D_{\mathcal{G}^{\downarrow}}(v)$. The fact that $\mathcal{G}'\langle C\rangle$ is the complete graph on vertices C then implies $F = D_{\mathcal{G}^{\downarrow}}(v)$ and $C = N_{\mathcal{G}^{\downarrow}}[v]$.

For the converse, we first show that $N_{\mathcal{G}^{\downarrow}}[v]$ is a maximal clique in $\mathcal{G}'$. By the definition of F, $N_{\mathcal{G}^{\downarrow}}[v]$ must be a clique of $\mathcal{G}'$, a clique not contained in $\mathcal{G}$, and therefore, $N_{\mathcal{G}^{\downarrow}}[v] \subseteq C$ for some new maximal clique C of $\mathcal{G}'$. Suppose $N_{\mathcal{G}^{\downarrow}}[v] \neq C$, so that there exists at least one vertex $v' \in C$,

$v' \notin N_{\mathcal{G}^\downarrow}[v]$. Since $\{v, v'\}$ is an edge in $\mathcal{G}'$ but $v' \notin N_{\mathcal{G}^\downarrow}[v]$, we must have $\{v, v'\} \in E$. Consider two cases: (i) $v' \neq \alpha(i)$ for $i = 1, \ldots, k$ and (ii) $v' = \alpha(i)$ for some $i = 1, \ldots, k$. In the first case, v' is a vertex in $\mathcal{G}^\downarrow$ and $\{v, v'\} \in E$; hence, $\{v, v'\}$ is an edge in $\mathcal{G}^\downarrow$. However, this contradicts the fact that $v' \notin N_{\mathcal{G}^\downarrow}[v]$. In the second case, let $v' = \alpha(i)$ for some $i = 1, \ldots, k$, and choose any edge $\{a, b\} \in F$. Since $\{v', a\}$ and $\{v', b\}$ are edges in $\mathcal{G}'$ and $v' \notin N_{\mathcal{G}^\downarrow}[v]$, we must have $\{v', a\} \in E$ and $\{v', b\} \in E$. Consequently, a and b are neighbors of v' in $\mathcal{G}$. Since $\{a, b\} \notin E$, the edge $\{a, b\}$ must be added at some point in the elimination process, and this contradicts the fact that α is a k-partial elimination ordering. Therefore, no such vertex v' exists, and $C = N_{\mathcal{G}^\downarrow}[v]$ is a maximal clique of $\mathcal{G}'$.

To show that $C = N_{\mathcal{G}^\downarrow}[v]$ is the unique new maximal clique contained in $\mathcal{G}'$, suppose there exists another maximal clique C' of $\mathcal{G}'$ not contained in $\mathcal{G}$. Then, there must be some edge $\{a, b\} \in F$ which formed this new maximal clique, and we must have $\{a, b\} \subset C$ and $\{a, b\} \subset C'$. Choose any vertex $v' \in C'$ with $v' \notin C$. Since $\{v', a\}$ and $\{v', b\}$ are edges in $\mathcal{G}'$ but $v' \notin N_{\mathcal{G}^\downarrow}[v]$, we must have $\{v', a\} \in E$ and $\{v', b\} \in E$. Since $\{a, b\} \in D_{\mathcal{G}^\downarrow}(v)$, we have $\{v, a\} \in E$ and $\{v, b\} \in E$, and in addition, $\{a, b\} \notin E$, and $\{v, v'\} \notin E$ since $v' \notin C$. This implies that $[v, a, v', b, v]$ is a chordless cycle in $\mathcal{G}$, which contradicts the fact that $\mathcal{G}$ is triangulated. Hence, no such v' exists, and C is the unique new maximal clique of $\mathcal{G}'$.

Finally, we show that $\mathcal{G}'$ is triangulated by providing a perfect elimination ordering for $\mathcal{G}'$. Since α is a k-partial elimination ordering for $\mathcal{G}$, it is also a k-partial elimination ordering for $\mathcal{G}'$, due to the fact that every edge $\{a, b\} \in F$ satisfies $\{a, b\} \subset V - \{\alpha(1), \ldots, \alpha(k)\}$ by definition. Next, using Lemma 2 in [89], adding edges $F = D_{\mathcal{G}^\downarrow}(v)$ to the triangulated graph $\mathcal{G}^\downarrow$ generates a new triangulated graph. Consequently, there exists a perfect elimination ordering β for this new graph, and the concatenation of $\alpha(1), \ldots, \alpha(k)$ with β gives a perfect elimination ordering for $\mathcal{G}'$. ■

■ B.4 Proof of Theorem 3.7

Theorem 3.7 (Conditional Independencies and Clique Extensions).
Let $\mathcal{G} = \mathcal{G}_0, \mathcal{G}_1, \mathcal{G}_2, \ldots, \mathcal{G}_n = \mathcal{G}'$ be a sequence of clique extensions with corresponding maximal cliques C_i as in Corollary 3.2. Then, for any density $p(x_V)$, the following are equivalent:

(1) $p_{\mathcal{G}'} = p_{\mathcal{G}}$,

(2) $p(x_{C_i}) = p_{\mathcal{G}_{i-1}}(x_{C_i}) = p_{\mathcal{G}_{i-1}(C_i)}(x)$ for $i = 1, \ldots, n$,

(3) X_{C_i} (under density p) is Markov with respect to the subgraph $\mathcal{G}_{i-1}(C_i)$ for $i = 1, \ldots, n$.

Proof. Notice that (2) directly implies (1) by using the relationship in (3.40). Suppose now that $p_{\mathcal{G}'} = p_{\mathcal{G}}$, and consider the decomposition proven in Proposition 3.24 which indicates

$$D\left(p_{\mathcal{G}'}(x) \| p_{\mathcal{G}}(x)\right) = \sum_{i=1}^{n} D\left(p(x_{C_i}) \| p_{\mathcal{G}_{i-1}(C_i)}(x_{C_i})\right).$$

Since $p_{\mathcal{G}'} = p_{\mathcal{G}}$, we must have $D\left(p_{\mathcal{G}'}(x) \| p_{\mathcal{G}}(x)\right) = 0$, and since the Kullback-Leibler divergence is always non-negative, we must have $D\left(p(x_{C_i}) \| p_{\mathcal{G}_{i-1}(C_i)}(x_{C_i})\right) = 0$ for $i = 1, \ldots, n$. This is true if and only if $p(x_{C_i}) = p_{\mathcal{G}_{i-1}(C_i)}(x_{C_i})$ almost everywhere. Therefore, (1) and (2) are equivalent.

To show that (2) and (3) are equivalent, recall that Theorem 3.2 does not require the positivity condition to be satisfied when a graph is a triangulated. Furthermore, notice that $\mathcal{G}_{i-1}(C_i)$

is a triangulated graph for $i = 1, \ldots, n$, since the induced subgraph of a triangulated graph is always triangulated [89]. Using Theorem 3.2, the density $p(x_{C_i})$ factors according to $\mathcal{G}_{i-1}(C_i)$, *i.e.* $p(x_{C_i}) = p_{\mathcal{G}_{i-1}(C_i)}(x)$, if and only if X_{C_i} is Markov with respect to $\mathcal{G}_{i-1}(C_i)$. ■

■ B.5 Proof of Proposition 3.10 and Corollary 3.3

Proposition 3.10 (Neighborhood Separator Coverings and Adding Edges).
Let $\mathcal{G} = (V, E)$ be a triangulated graph with neighborhood separator covering $\{S_i\}_{i=1}^m$. Given some $v \in V$, define $\mathcal{G}' \triangleq (V, E \cup D_{\mathcal{G}}(v))$. Then, $\{S_i\}_{i=1}^m$ is also a neighborhood separator covering for $\mathcal{G}'$.

Proof. Suppose vertex v is an element of the separator S_i. There are three cases to consider.

(1) By Proposition 3.7, $\mathcal{G}'$ is a clique extension with new maximal clique $N_{\mathcal{G}}[v]$, but since $v \in S_i$ and S_i is a neighborhood separator of $\mathcal{G}$, we have $N_{\mathcal{G}}[v] = N_{\mathcal{G}}[S_i]$. Hence, $N_{\mathcal{G}}[S_i]$ is a maximal clique in $\mathcal{G}'$, which implies that S_i is a trivial neighborhood separator for $\mathcal{G}'$.

(2) Consider the neighborhood separator $S_j \neq S_i$. If there is no edge $\{a, b\} \in D_{\mathcal{G}}(v)$ with $\{a, b\} \subset N_{\mathcal{G}}[S_j]$, then the structure of the subgraph induced by $N_{\mathcal{G}}[S_j]$ is the same in $\mathcal{G}$ and $\mathcal{G}'$. Hence, S_j is a neighborhood separator for $\mathcal{G}'$.

(3) Consider the neighborhood separator $S_j \neq S_i$, and suppose there exists an edge $\{a, b\} \in D_{\mathcal{G}}(v)$ with $\{a, b\} \subset N_{\mathcal{G}}[S_j]$. We now show that S_j satisfies the first property of a neighborhood separator. Choose any $v_i \in S_i$ and $v_j \in S_j$, and note that $[v_i, a, v_j, b, v_i]$ is a cycle in $\mathcal{G}$. Since $\mathcal{G}$ is triangulated and since $\{a, b\} \notin E$, we must have $\{v_i, v_j\} \in E$. This is true for all $v_i \in S_i$ and $v_j \in S_j$, and hence $S_j \subset N_{\mathcal{G}}[S_i]$. Recall that $\mathcal{G}'$ is a clique extension with new maximal clique $N_{\mathcal{G}}[S_i]$; therefore, in the graph $\mathcal{G}'$, each vertex $v_j \in S_j$ is a neighbor of every vertex in $N_{\mathcal{G}}[S_i]$. This implies that $N_{\mathcal{G}'}[v_j] = N_{\mathcal{G}}[v_j] \cup N_{\mathcal{G}}[S_i]$, and since v_j is an element of the neighborhood separator S_j, we have $N_{\mathcal{G}'}[v_j] = N_{\mathcal{G}}[S_j] \cup N_{\mathcal{G}}[S_i]$. Notice that $N_{\mathcal{G}'}[v_j]$ is the same regardless of the choice of $v_j \in S_j$, and consequently, $N_{\mathcal{G}'}[v_j] = N_{\mathcal{G}'}[S_j]$ for all $v_j \in S_j$.

In order to prove that S_j satisfies the second requirement of a neighborhood separator, notice that every $\{a, b\} \in D_{\mathcal{G}}(v)$ must satisfy $\{a, b\} \subset N_{\mathcal{G}}[S_i]$. Therefore, the addition of the edges in $D_{\mathcal{G}}(v)$ does not change the structure induced by vertices $N_{\mathcal{G}}[S_j] - N_{\mathcal{G}}[S_i]$ in the graph $\mathcal{G}'$, and in addition, $N_{\mathcal{G}}[S_i]$ is a clique in $\mathcal{G}'$. Thus, S_j is a neighborhood separator if we can show that there is no edge $\{v, v'\}$ in $\mathcal{G}$ with $v \in N_{\mathcal{G}}[S_i] - S_j$ and $v' \in N_{\mathcal{G}}[S_j] - N_{\mathcal{G}}[S_i]$. First, if $v \in S_i$, then $\{v, v'\} \notin E$ because $v' \notin N_{\mathcal{G}}[S_i]$. Similarly, if $v \in N_{\mathcal{G}}[S_i] - N_{\mathcal{G}}[S_j]$, then $\{v, v'\} \notin E$ because $v \notin N_{\mathcal{G}}[S_j]$. Now, let $v \in N_{\mathcal{G}}[S_i] \cap N_{\mathcal{G}}[S_j]$, $v \notin S_i$, $v \notin S_j$. Choose any $v_i \in S_i$, $v_j \in S_j$, and note that the following edges are present in $\mathcal{G}$: $\{v_i, v\}$, $\{v_j, v\}$, $\{v_j, v_i\}$, $\{v_j, v'\}$. Suppose that $\{v, v'\} \in E$, then since S_j is a neighborhood separator for $\mathcal{G}$, we must have $\{v_i, v'\} \in E$, but this contradicts the fact that $v' \notin N_{\mathcal{G}}[S_i]$. Hence, no such edge $\{v, v'\}$ exists in $\mathcal{G}$, thereby proving the result.

■

Corollary 3.3 (Neighborhood Separator Coverings and Vertex Elimination).
Let $\mathcal{G} = (V, E)$ be a triangulated graph with neighborhood separator covering $\{S_i\}_{i=1}^m$. Given some $v \in V$ and $v \in S_i$, define $\mathcal{G}^{\downarrow} \triangleq \downarrow\!(\mathcal{G}, v)$. Then, $\{S_1, \ldots, S_{i-1}, S_i - \{v\}, S_{i+1}, \ldots, S_m\}$ is a neighborhood separator covering for $\mathcal{G}^{\downarrow}$.

Proof. From Proposition 3.10, we know that $\{S_i\}$ is a neighborhood separator covering for $\mathcal{G}' = (V, E \cup D_{\mathcal{G}}(v))$. Now, we only need to show that $\{S_1, \ldots, S_{i-1}, S_i - \{v\}, S_{i+1}, \ldots, S_m\}$ is a neighborhood separator covering for $\mathcal{G}^{\downarrow} = \mathcal{G}'(V - \{v\})$. There are three cases to consider.

(1) Suppose $v \notin N_{\mathcal{G}'}[S_j]$ for some $S_j \neq S_i$. Then, the neighborhood structure of S_j remains the same in $\mathcal{G}'$ and $\mathcal{G}^{\downarrow}$, and S_j is also a neighborhood separator for $\mathcal{G}^{\downarrow}$.

(2) Suppose $v \in N_{\mathcal{G}'}[S_j]$ for some $S_j \neq S_i$. Since we are only removing a vertex from the graph and not introducing additional edges, the neighborhood structure of $N_{\mathcal{G}^{\downarrow}}[S_j] = N_{\mathcal{G}'}[S_j] - \{v\}$ does not change, and therefore, S_j is also a neighborhood separator for $\mathcal{G}^{\downarrow}$.

(3) If $S_i - \{v\} = \emptyset$, then we are done. Otherwise, define $S \triangleq S_i - \{v\}$. Notice that for each $v' \in S$, we have $N_{\mathcal{G}^{\downarrow}}[v'] = N_{\mathcal{G}}[S_i] - \{v\} = N_{\mathcal{G}}[S]$, and hence, the first property of a neighborhood separator is satisfied. The set S also satisfies the second property of a neighborhood separator since removing vertex v does not change the neighborhood structure of $N_{\mathcal{G}^{\downarrow}}[S]$ in $\mathcal{G}^{\downarrow}$.

■

■ B.6 Proof of Proposition 3.11

Proposition 3.11 (Edges Associated with the Modified Elimination Game).
Let $\mathcal{G} = (V, E)$ be an arbitrary graph, and let $\widetilde{\mathcal{G}}_i = (V, E_i)$ be defined according to (3.48) for some ordering α and some $M \subseteq V$. Define β^{-1} according to (3.49). Then, $\{a, b\} \in E_i$ if and only if there exists a path $[a = v_1, v_2, \ldots, v_k, v_{k+1} = b]$ in $\mathcal{G}$ such that $\alpha^{-1}(v_j) < \min(\beta^{-1}(a), \beta^{-1}(b), i+1)$, for $j = 2, \ldots, k$.

Proof. We show that such a path always exists by using induction on i. First, suppose $\{a, b\} \in E_0$. Then, $\{a, b\} \in E$, and there exists a trivial path in $\mathcal{G}$, *i.e.* $k = 1$ in this case. Suppose now the result holds for all $i < i_0$, and consider the case $i = i_0$. Let $\{a, b\} \in E_i$, in which case, either $\{a, b\} \in E_{i-1}$ or $\{a, b\} \in \widetilde{D}^{\downarrow}_{\mathcal{G}}(\alpha(i))$. If $\{a, b\} \in E_{i-1}$, the induction hypothesis provides the needed path.

Suppose $\{a, b\} \in \widetilde{D}^{\downarrow}_{\mathcal{G}}(\alpha(i))$. Then, $\{a, b\} \subset N_{\widetilde{\mathcal{G}}^{\downarrow}_{i-1}}(\alpha(i))$, which implies $\{a, \alpha(i)\} \in E_{i-1}$ and $\{b, \alpha(i)\} \in E_{i-1}$. By the induction hypothesis, the following two paths exist in $\mathcal{G}$:

(1) $[a = y_1, y_2, \ldots, y_l, y_{l+1} = \alpha(i)]$ with $\alpha^{-1}(y_j) < \min\left(\beta^{-1}(a), \beta^{-1}(\alpha(i)), i\right)$, $j = 2, \ldots, l$

(2) $[\alpha(i) = z_1, z_2, \ldots, z_m, z_{m+1} = b]$ with $\alpha^{-1}(z_j) < \min\left(\beta^{-1}(\alpha(i)), \beta^{-1}(b), i\right)$, $j = 2, \ldots, m$.

Consider the walk $[a = y_1, y_2, \ldots, y_l, \alpha(i), z_2, \ldots, z_m, z_{m+1} = b]$ in $\mathcal{G}$. By the definition of β^{-1}, the condition $i \leq \beta^{-1}(\alpha(i))$ is always satisfied, and so, we can simplify the inequalities in (1) and (2) above as follows,

$$\alpha^{-1}(y_j) < \min\left(\beta^{-1}(a), i\right), \quad j = 2, \ldots, l \tag{B.5a}$$

$$\alpha^{-1}(z_j) < \min\left(\beta^{-1}(b), i\right), \quad j = 2, \ldots, m. \tag{B.5b}$$

Since $\{a,b\} \in \widetilde{D}^{\downarrow}_{\mathcal{G}}(\alpha(i))$, a and b are by definition contained in the elimination graph $\widetilde{\mathcal{G}}^{\downarrow}_{i-1}$. This implies that $\beta^{-1}(a) > i$ and $\beta^{-1}(b) > i$, which may also be written as follows,

$$\alpha^{-1}\left(\alpha(i)\right) = i < \min\left(\beta^{-1}(a), \beta^{-1}(b), i+1\right). \tag{B.6}$$

Using (B.6) in conjunction with (B.5) gives

$$\alpha^{-1}(y_j) < \min\left(\beta^{-1}(a), \beta^{-1}(b), i+1\right), \quad j = 2, \ldots, l \tag{B.7a}$$

$$\alpha^{-1}(z_j) < \min\left(\beta^{-1}(a), \beta^{-1}(b), i+1\right), \quad j = 2, \ldots, m. \tag{B.7b}$$

The inequalities in (B.6) and (B.7) are precisely the inequalities needed for each vertex in the walk $[a = y_1, y_2, \ldots, y_l, \alpha(i), z_2, \ldots, z_m, z_{m+1} = b]$. If the walk is not a path, it can be turned into one by removing the appropriate repeated vertices.

To prove the other direction, we also use induction on i. Let $i = 0$, and suppose there exists a path $[a = v_1, v_2, \ldots, v_k, v_{k+1} = b]$ in $\mathcal{G}$ such that $\alpha^{-1}(v_j) < \min(\beta^{-1}(a), \beta^{-1}(b), 1)$ for $j = 2, \ldots, k$. Since each $\alpha^{-1}(v_j) \geq 1$, we must have $k = 1$ and $[a, b]$ a path in $\mathcal{G}$. This implies $\{a,b\} \in E = E_0$.

Suppose the converse holds for each $i < i_0$, and consider the case $i = i_0$. Suppose there exists a path $[a = v_1, v_2, \ldots, v_k, v_{k+1} = b]$ in $\mathcal{G}$ satisfying $\alpha^{-1}(v_j) < \min(\beta^{-1}(a), \beta^{-1}(b), i+1)$ for $j = 2, \ldots, k$. Define $v' \triangleq v_t$ where $\alpha^{-1}(v_t) = \max\{\alpha^{-1}(v_j) | 2 \leq j \leq k\}$, and define $m \triangleq \alpha^{-1}(v_t)$. Therefore, each $v_j \neq v'$, $j = 2, \ldots, k$, satisfies $\alpha^{-1}(v_j) < m = \alpha^{-1}(v') < \min(\beta^{-1}(a), \beta^{-1}(b), i+1)$, and since $\alpha^{-1}(v') \leq \beta^{-1}(v')$, this gives $\alpha^{-1}(v_j) < \min(\beta^{-1}(a), \beta^{-1}(b), \beta^{-1}(v'), m)$.

Consider now the two paths $[a = v_1, v_2, \ldots, v_{t-1}, v']$ and $[v', v_{t+1}, \ldots, v_k, v_{k+1} = b]$ in $\mathcal{G}$, where the following inequalities are satisfied,

$$\alpha^{-1}(v_j) < \min(\beta^{-1}(a), \beta^{-1}(v'), m), \quad j = 2, \ldots, t-1 \tag{B.8a}$$

$$\alpha^{-1}(v_j) < \min(\beta^{-1}(v'), \beta^{-1}(b), m), \quad j = t+1, \ldots, k. \tag{B.8b}$$

Since $m < i+1$, the induction hypothesis gives $\{a, v'\} \in E_{m-1}$ and $\{v', b\} \in E_{m-1}$, and consequently, we know that $\{a,b\} \subseteq N_{\widetilde{\mathcal{G}}_{m-1}}(v')$. Using the inequality $\alpha^{-1}(v') < \min(\beta^{-1}(a), \beta^{-1}(b), i+1)$ implies that vertices a and b, in addition to v', are in the elimination graph $\widetilde{\mathcal{G}}^{\downarrow}_{m-1}$. Therefore, $\{a,b\} \subseteq \widetilde{N}^{\downarrow}_{\mathcal{G}}(v')$, and either $\{a,b\}$ is an edge in the elimination graph $\widetilde{\mathcal{G}}^{\downarrow}_{m-1}$ or $\{a,b\} \in \widetilde{D}^{\downarrow}_{\mathcal{G}}(v')$. In either case, $\{a,b\} \in E_m$ and therefore $\{a,b\} \in E_i$. ■

■ B.7 Proof of Propositions 3.12 and 3.14

Before proving Propositions 3.12 and 3.14, we state an important lemma concerning the relationship between the graphs $\widetilde{\mathcal{G}}^{\downarrow}_i$ in (3.45) and $\widetilde{\mathcal{G}}_i$ in (3.48). Specifically, the graph $\widetilde{\mathcal{G}}^{\downarrow}_i$ may be generated from $\widetilde{\mathcal{G}}_i$ by eliminating all vertices $\alpha(j) \notin M$ with $j \leq i$, *i.e.* the vertices in the set A_i,

$$A_0 \triangleq \emptyset \tag{B.9a}$$

$$A_i \triangleq \{\alpha(j) | j \leq i, \alpha(j) \notin M\}, \quad i = 1, \ldots, n. \tag{B.9b}$$

Defining $k_i \triangleq |A_i|$, we also show in Lemma B.1 that a k_i-partial elimination ordering down to the set $V - A_i$ exists for the graph $\widetilde{\mathcal{G}}_i$.

Lemma B.1 ($\widetilde{\mathcal{G}}_i^{\downarrow}$ is an Elimination Graph of $\widetilde{\mathcal{G}}_i$).
Let the sequence of graphs $\widetilde{\mathcal{G}}_i^{\downarrow}$ and $\widetilde{\mathcal{G}}_i$, $i = 0, \ldots, n$ be defined according to (3.45) and (3.48) respectively for a given graph $\mathcal{G} = (V, E)$, an ordering α on V, and a set $M \subseteq V$. If A_i is defined according to (B.9), then there exists a k_i-partial elimination ordering $\bar{\alpha}$ down to $V - A_i$ for the graph $\widetilde{\mathcal{G}}_i$ such that $\widetilde{\mathcal{G}}_i^{\downarrow} = \widetilde{\mathcal{G}}_i \langle V - \{\bar{\alpha}(1), \ldots, \bar{\alpha}(k_i)\}\rangle = \widetilde{\mathcal{G}}_i \langle V - A_i \rangle = \widetilde{\mathcal{G}}_i (V - A_i)$ for $i = 0, \ldots, n$.

Proof. By definition, $\widetilde{\mathcal{G}}_0^{\downarrow} = \mathcal{G} = \widetilde{\mathcal{G}}_0 \langle V - A_0 \rangle = \widetilde{\mathcal{G}}_0$, and since $A_0 = \emptyset$, any ordering on the vertices V is a 0-partial ordering for $\widetilde{\mathcal{G}}_0$. Assume that the result holds for $i < i_0$, and consider the case $i = i_0$. Let $\bar{\alpha}$ be a k_{i-1}-partial elimination ordering of $\widetilde{\mathcal{G}}_{i-1}$ down to $V - A_{i-1}$ such that $\widetilde{\mathcal{G}}_{i-1}^{\downarrow} = \widetilde{\mathcal{G}}_{i-1} \langle V - \{\bar{\alpha}(1), \ldots, \bar{\alpha}(k_{i-1})\}\rangle = \widetilde{\mathcal{G}}_{i-1} \langle V - A_{i-1} \rangle$.

Using (3.48), $\widetilde{\mathcal{G}}_i = (V, E_i) = (V, E_{i-1} \cup F)$ where $F = \widetilde{D}_{\mathcal{G}}^{\downarrow}(\alpha(i)) = D_{\widetilde{\mathcal{G}}_{i-1}^{\downarrow}}(\alpha(i))$. Consider now the elimination graph $\mathcal{G}^{\downarrow} \triangleq \widetilde{\mathcal{G}}_i \langle V - \{\bar{\alpha}(1), \ldots, \bar{\alpha}(k_{i-1})\}\rangle$, and suppose $\widetilde{\mathcal{G}}_{i-1}^{\downarrow} = (V_{i-1}, F_{i-1})$ as in (3.45). Since F only contains vertices in the elimination graph $\widetilde{\mathcal{G}}_{i-1}^{\downarrow}$ and since the vertices $\bar{\alpha}(1), \ldots, \bar{\alpha}(k_{i-1})$ are not in $\widetilde{\mathcal{G}}_{i-1}^{\downarrow}$, we must have $\mathcal{G}^{\downarrow} = (V_{i-1}, F_{i-1} \cup F)$, and in addition, $\bar{\alpha}$ is a k_{i-1}-partial elimination ordering for $\widetilde{\mathcal{G}}_i$ down to $V - A_{i-1}$.

Consider now the two possible cases for $\alpha(i)$. If $\alpha(i) \in M$ then $A_i = A_{i-1} = \{\bar{\alpha}(1), \ldots, \bar{\alpha}(k_{i-1})\}$ and $k_i = k_{i-1}$. By the definition in (3.45), $\widetilde{\mathcal{G}}_i^{\downarrow} = (V_{i-1}, F_{i-1} \cup F) = \mathcal{G}^{\downarrow} = \widetilde{\mathcal{G}}_i \langle V - A_i \rangle$, and $\bar{\alpha}$ is a k_i-partial elimination ordering for $\widetilde{\mathcal{G}}_i$ down to $V - A_i = V - A_{i-1}$. If $\alpha(i) \notin M$ then $A_i = A_{i-1} \cup \{\alpha(i)\} = \{\bar{\alpha}(1), \ldots, \bar{\alpha}(k_{i-1}), \alpha(i)\}$. Using (3.45), $\widetilde{\mathcal{G}}_i^{\downarrow} = \downarrow\left(\widetilde{\mathcal{G}}_{i-1}^{\downarrow}, \alpha(i)\right)$, and since $\mathcal{G}^{\downarrow} = (V_{i-1}, F_{i-1} \cup F)$, we can also write $\widetilde{\mathcal{G}}_i^{\downarrow} = \downarrow(\mathcal{G}^{\downarrow}, \alpha(i))$ which in turn is equal to $\widetilde{\mathcal{G}}_i^{\downarrow} = \downarrow\left(\widetilde{\mathcal{G}}_i \langle V - \{\bar{\alpha}(1), \ldots, \bar{\alpha}(k_{i-1})\}\rangle, \alpha(i)\right) = \widetilde{\mathcal{G}}_i \langle V - A_i \rangle$. Furthermore, since $\mathcal{G}^{\downarrow}$ contains the edges in F, vertex $\alpha(i)$ may be eliminated from $\mathcal{G}^{\downarrow}$ without introducing fill edges, and consequently, any ordering of the form $(\bar{\alpha}(1), \ldots, \bar{\alpha}(k_{i-1}), \alpha(i), \ldots)$. is a k_i-partial elimination ordering for $\widetilde{\mathcal{G}}_i$ down to $V - A_i$.

Finally, since there exists a k_i-partial elimination ordering of $\widetilde{\mathcal{G}}_i$ down to $V - A_i$, Lemma 3.5 indicates that $\widetilde{\mathcal{G}}_i \langle V - A_i \rangle = \widetilde{\mathcal{G}}_i (V - A_i)$. ■

Proposition 3.12 (Modified Elimination Game and Clique Extensions).
Let $\mathcal{G} = (V, E)$ be a triangulated graph. Given any set $M \subseteq V$ and any ordering α on the vertices V, the sequence of graphs $\widetilde{\mathcal{G}}_i$ in (3.48) form a sequence of clique extensions, and the new maximal clique C_i contained in $\widetilde{\mathcal{G}}_i$ but not $\widetilde{\mathcal{G}}_{i-1}$ is given by $C_i = \widetilde{N}_{\mathcal{G}}^{\downarrow}[\alpha(i)]$.

Proof. We prove the result by induction. By assumption, the initial graph $\widetilde{\mathcal{G}}_0 = \mathcal{G}$ is triangulated. Assume now that $\widetilde{\mathcal{G}}_{i-1}$ is triangulated. By applying Proposition 3.7, we show that $\widetilde{\mathcal{G}}_i$ is a clique extension of $\widetilde{\mathcal{G}}_{i-1}$.

Set $F = D_{\widetilde{\mathcal{G}}_{i-1}^{\downarrow}}(\alpha(i)) = \widetilde{D}_{\mathcal{G}}^{\downarrow}(\alpha(i))$, and recall from (3.48b) that $\widetilde{\mathcal{G}}_i = (V, E_i) = (V, E_{i-1} \cup F)$. According to Lemma B.1, there exists a k_{i-1}-partial elimination ordering $\bar{\alpha}$ of $\widetilde{\mathcal{G}}_{i-1}$ such that $\widetilde{\mathcal{G}}_{i-1}^{\downarrow} = \widetilde{\mathcal{G}}_{i-1} \langle V - \{\bar{\alpha}(1), \ldots, \bar{\alpha}(k_{i-1})\rangle = \widetilde{\mathcal{G}}_{i-1} (V - A_{i-1})$. Since $\widetilde{\mathcal{G}}_{i-1}^{\downarrow}$ is an induced subgraph of $\widetilde{\mathcal{G}}_{i-1}$, the set $F = D_{\widetilde{\mathcal{G}}_{i-1}^{\downarrow}}(\alpha(i))$ satisfies $F \subseteq D_{\widetilde{\mathcal{G}}_{i-1}}(\alpha(i))$. Therefore, F, $\widetilde{\mathcal{G}}_{i-1}$, and $\widetilde{\mathcal{G}}_i$ satisfy all of the requirements of Proposition 3.7, thereby implying that $\widetilde{\mathcal{G}}_i$ is a clique extension of $\widetilde{\mathcal{G}}_{i-1}$. Furthermore, Proposition 3.7 indicates that $C_i = N_{\widetilde{\mathcal{G}}_{i-1}^{\downarrow}}[\alpha(i)] = \widetilde{N}_{\mathcal{G}}^{\downarrow}[\alpha(i)]$ is the unique maximal clique contained in $\widetilde{\mathcal{G}}_i$ but not $\widetilde{\mathcal{G}}_{i-1}$. ■

Proposition 3.14 (Modified Elimination Game and Theorem 3.7).
Suppose $\mathcal{G} = (V, E)$ is a triangulated graph with a neighborhood separator covering. Let the sequence of graphs $\widetilde{\mathcal{G}}_i$ be defined according to (3.48) for some set $M \subseteq V$ and an ordering α on V. Let $\mathcal{C}_i$ denote the set of maximal cliques of the subgraph $\widetilde{\mathcal{G}}_{i-1}(C_i)$, where $C_i \triangleq \widetilde{N}^{\downarrow}_{\mathcal{G}}[\alpha(i)]$ for $i = 1 \dots n$. Then, for any density $p(x_V)$, the following are equivalent:

(1) $p_{\widetilde{\mathcal{G}}_i} = p_{\mathcal{G}}$ for some $1 \leq i \leq n$,

(2) the conditions $\perp X_{\mathcal{C}_j}$ are satisfied (under density p) for all $1 \leq j \leq i$.

Proof. Choose some $1 \leq i \leq n$, and consider the sequence $\widetilde{\mathcal{G}}_0 = \mathcal{G}, \widetilde{\mathcal{G}}_1, \dots, \widetilde{\mathcal{G}}_i$, which by Proposition 3.12 is a sequence of clique extensions with corresponding new maximal cliques $C_j = \widetilde{N}^{\downarrow}_{\mathcal{G}}[\alpha(j)]$, $1 \leq j \leq i$. Using Theorem 3.7, $p_{\widetilde{\mathcal{G}}_i} = p_{\mathcal{G}}$ if and only if X_{C_j} is Markov with respect to the subgraph $\widetilde{\mathcal{G}}_{j-1}(C_j)$ for $1 \leq j \leq i$.

We first show that $\widetilde{\mathcal{G}}_{j-1}(C_j) = \widetilde{\mathcal{G}}^{\downarrow}_{j-1}(C_j)$. Using Lemma B.1, $\widetilde{\mathcal{G}}^{\downarrow}_{j-1} = \widetilde{\mathcal{G}}_{j-1}\langle V - A_{j-1}\rangle = \widetilde{\mathcal{G}}_{j-1}(V - A_{j-1})$. Furthermore, the set $C_j = \widetilde{N}^{\downarrow}_{\mathcal{G}}[\alpha(j)] = N_{\widetilde{\mathcal{G}}^{\downarrow}_{j-1}}[\alpha(j)]$ is a subset of the vertices in the elimination graph $\widetilde{\mathcal{G}}^{\downarrow}_{j-1}$, *i.e.* $C_j \subseteq V - A_{j-1}$. Then, $\widetilde{\mathcal{G}}^{\downarrow}_{j-1}(C_j)$ is an induced subgraph of $\widetilde{\mathcal{G}}^{\downarrow}_{j-1}$ which in turn is an induced subgraph of $\widetilde{\mathcal{G}}_{j-1}$, and therefore, we must have $\widetilde{\mathcal{G}}^{\downarrow}_{j-1}(C_j) = \widetilde{\mathcal{G}}_{j-1}(C_j)$.

By Proposition 3.13, the graph $\widetilde{\mathcal{G}}^{\downarrow}_{j-1}$ has a neighborhood separator covering, and therefore, vertex $\alpha(j) \in C_j$ is a subset of some neighborhood separator S in $\widetilde{\mathcal{G}}^{\downarrow}_{j-1}$. By the definition of a neighborhood separator, $C_j = N_{\widetilde{\mathcal{G}}^{\downarrow}_{j-1}}[\alpha(j)] = N_{\widetilde{\mathcal{G}}^{\downarrow}_{j-1}}[S]$, and consequently, the process X_{C_j} is Markov with respect to $\widetilde{\mathcal{G}}_{j-1}(C_j) = \widetilde{\mathcal{G}}^{\downarrow}_{j-1}(C_j)$ if and only if the conditions $\perp X_{\mathcal{C}_j}$ are satisfied. ■

■ B.8 Proof of Proposition 3.16

Before proving Proposition 3.16, we provide two important lemmas. Recall that the boundary sets B_{v_i} are defined in (2.22) for $i = 2, \dots, m$. In order to simplify subsequent statements as well as some of the inductive proofs, it is useful to define $B_{v_1} \triangleq V$, *i.e.* the first boundary set contains all vertices in the graph. The first lemma provides an alternative characterization of the boundary sets which lends itself more easily to the graph-theoretic arguments to follow.

Lemma B.2 (Alternative Characterization of Boundary Sets B_{v_i}).
Let $(v_1, \dots, v_m)$ be an ordering on the non-leaf vertices of a rooted tree $\mathcal{G}_{\preceq} = (V, E)$, and let α be a leaf-last ordering on V satisfying (3.50). The boundary sets B_{v_i}, $i = 1, \dots, m$, may equivalently be characterized as follows,

$$B_{v_i} = \{v_i\} \cup \big\{v \in V \,\big|\, \textit{there exists a path } [v = u_1, u_2, \dots, u_k, u_{k+1} = v_i] \textit{ in } \mathcal{G}^{\sim}_{\preceq} \\ \textit{such that } \alpha^{-1}(u_j) < \alpha^{-1}(v_i),\ 2 \leq j \leq k\big\}. \tag{B.10}$$

Proof. For $i = 1, \dots, m$, we define the sets $\tilde{B}_{v_i}$ as in (B.10), *i.e.*

$$\tilde{B}_{v_i} = \{v_i\} \cup \big\{v \in V \,\big|\, \text{there exists a path } [v = u_1, u_2, \dots, u_k, u_{k+1} = v_i] \text{ in } \mathcal{G}^{\sim}_{\preceq} \\ \text{such that } \alpha^{-1}(u_j) < \alpha^{-1}(v_i),\ 2 \leq j \leq k\big\},$$

and we show that $\tilde{B}_{v_i} = B_{v_i}$. Consider separately the case of v_1, and recall that $B_{v_1} = V$. By definition, $\tilde{B}_{v_1} \subset V$ and $v_1 \in \tilde{B}_{v_1}$. Now, choose any $v \in V$, $v \neq v_1$, and notice that there exists a unique path $[v = u_1, u_2, \ldots, u_k, u_{k+1} = v_1]$ in $\mathcal{G}_{\precsim}^{\sim}$ between v and v_1. Any such path must have all u_j, $2 \leq j \leq k$, equal to non-leaf vertices distinct from v_1, and as a result, $\alpha^{-1}(u_j) < \alpha^{-1}(v_1) = m$ by the definition of α. Thus, $V \subset \tilde{B}_{v_1}$, thereby proving $\tilde{B}_{v_1} = B_{v_1} = V$.

Choose any v_i, $i = 2, \ldots, m$. By definition, $v_i \in B_{v_i}$ and $v_i \in \tilde{B}_{v_i}$. Choose any $v^* \in B_{v_i}$ with $v^* \neq v_i$, and notice that there exists a unique path $[v^* = u_1, u_2, \ldots, u_k, u_{k+1} = v_i]$ in $\mathcal{G}_{\precsim}^{\sim}$ between v^* and v_i, where each u_j, $2 \leq j \leq k$, is a non-leaf vertex distinct from v_i. Since $B_{v_i} = \bigcap_{v < v_i} S_v^{v_i}$, we must have $v^* \in S_v^{v_i}$ for all $v < v_i$, meaning that v^* is in the same subtree $S_v^{v_i}$ as v_i. This implies that each u_j, $2 \leq j \leq k$, cannot satisfy $u_j = v$ for some vertex $v < v_i$. Therefore, each u_j must be a non-leaf vertex with $u_j > v_i$ for $2 \leq j \leq k$, which by the definition of α is the same as $\alpha^{-1}(u_j) < \alpha^{-1}(v_i)$. Hence, $v^* \in \tilde{B}_{v_i}$ and $B_{v_i} \subset \tilde{B}_{v_i}$.

Similarly, for any $i = 2, \ldots, m$, choose $v^* \in \tilde{B}_{v_i}$ with $v^* \neq v_i$, so that there exists a path $[v^* = u_1, u_2, \ldots, u_k, u_{k+1} = v_i]$ in $\mathcal{G}_{\precsim}^{\sim}$ with $\alpha^{-1}(u_j) < \alpha^{-1}(v_i)$, $2 \leq j \leq k$. Thus, each u_j, $2 \leq j \leq k$, is a non-leaf vertex satisfying $u_j > v_i$, implying that $u_j \neq v$ for any $v < v_i$. This in turn implies that v^* is in the same subtree $S_v^{v_i}$ as v_i for all $v < v_i$, and consequently, $v^* \in B_{v_i}$, thereby proving $\tilde{B}_{v_i} \subset B_{v_i}$. ■

The following lemma provides several useful properties of the subgraph $\widetilde{\mathcal{G}}_{m-i}(C_{m-i+1})$ for $i = 1, \ldots, m$. In the second property, we use the set $M^{(i)}$; recall that this set is associated with the marginalization-invariant Markov property and is defined in (2.31).

Lemma B.3 (Characterization of the Subgraphs $\widetilde{\mathcal{G}}_{m-i}(C_{m-i+1})$).

Let $(v_1, \ldots, v_m)$ be an ordering on the non-leaf vertices of a rooted tree $\mathcal{G}_{\precsim} = (V, E)$, and let α be a leaf-last ordering on V satisfying (3.50). Suppose $M \subseteq V$ contains all the leaf vertices of $\mathcal{G}_{\precsim}$, and let $\widetilde{\mathcal{G}}_i$ be generated by the modified elimination game according to (3.48) with $\widetilde{\mathcal{G}}_0 = \mathcal{G}_{\precsim}^{\sim}$. Denote the unique maximal clique contained in $\widetilde{\mathcal{G}}_i$ but not $\widetilde{\mathcal{G}}_{i-1}$ by C_i. Then, for $i = 1, \ldots, m$ the following three items are true:

(1) $\alpha(m - i + 1) = v_i$ is a neighborhood separator in the subgraph $\widetilde{\mathcal{G}}_{m-i}(C_{m-i+1})$.

(2) $C_{m-i+1} = B_{v_i} \cap M^{(i)}$.

(3) Edge $\{v, v'\}$ is present in the subgraph $\widetilde{\mathcal{G}}_{m-i}(C_{m-i+1})$ if and only if either $v' = v_i$ or the unique path between v and v' in $\mathcal{G}_{\precsim}^{\sim}$ does not pass through v_i.

Proof.

(1) By the definition of α, $v_i = \alpha(m - i + 1)$, and using Proposition 3.12, $C_{m-i+1} = \widetilde{N}^{\downarrow}_{\mathcal{G}_{\precsim}^{\sim}}[\alpha(m - i + 1)] = N_{\widetilde{\mathcal{G}}^{\downarrow}_{m-i}}[v_i]$ is the neighborhood of v_i in the elimination graph $\widetilde{\mathcal{G}}^{\downarrow}_{m-i}$. Recall from Lemma B.1 that $\widetilde{\mathcal{G}}^{\downarrow}_{m-i} = \widetilde{\mathcal{G}}_{m-i}(V - A_{m-i})$. Since the collection of vertices $\{\{v\}\}_{v \in V}$ form a neighborhood separator covering for $\widetilde{\mathcal{G}}_0 = \mathcal{G}_{\precsim}^{\sim}$, Proposition 3.10 and Corollary 3.3 may be used along with induction to show that $\{\{v\}\}_{v \in V - A_{m-i}}$ is a neighborhood separator covering for the elimination graph $\widetilde{\mathcal{G}}^{\downarrow}_{m-i}$. Therefore, $\alpha(m - i + 1) = v_i \in V - A_{m-i}$ is a neighborhood separator in the graph $\widetilde{\mathcal{G}}^{\downarrow}_{m-i}$, with a neighborhood equal to C_{m-i+1}, and $\widetilde{\mathcal{G}}_{m-i}(C_{m-i+1})$ is the subgraph of $\widetilde{\mathcal{G}}_{m-i}$ induced by the neighborhood of a neighborhood separator. Thus, $\alpha(m - i + 1) = v_i$ is a neighborhood separator in $\widetilde{\mathcal{G}}_{m-i}(C_{m-i+1})$.

(2) We first show that $C_{m-i+1} \subseteq M^{(i)}$ for $i = 1, \ldots, m$. By definition, the elements of C_{m-i+1} are vertices in the elimination graph $\tilde{\mathcal{G}}^{\downarrow}_{m-i} = \tilde{\mathcal{G}}_{m-i}(V - A_{m-i})$, and consequently, $C_{m-i+1} \subseteq V - A_{m-i}$. Consider separately the case $i = m$, in which case $V - A_{m-i} = V - A_0 = V$ and $C_{m-i+1} = C_1 \subseteq V = M^{(m)}$. Using (B.9b) for $i = 1, \ldots, m-1$, the set $V - A_{m-i}$ may be characterized as follows,

$$V - A_{m-i} = V - \{\alpha(j) | j \leq m - i, \alpha(j) \notin M\} = M \cup \{\alpha(j) | j > m - i\}.$$

Since M contains all the leaf vertices of $\mathcal{G}_{\preceq}$, this set may equivalently be written as follows,

$$V - A_{m-i} = M \cup \{\alpha(j) | j \geq m - i + 1\} = M \cup \{v | v \leq v_i\} = M^{(i)}.$$

Hence, $C_{m-i+1} \subseteq V - A_{m-i} = M^{(i)}$.

By Proposition 3.12, C_{m-i+1} is a maximal clique in $\tilde{\mathcal{G}}_{m-i+1}$, and it is the unique maximal clique in $\tilde{\mathcal{G}}_{m-i+1}$ containing v_i. Consequently, the set C_{m-i+1} contains v_i plus all vertices v such that $\{v, v_i\} \in E_{m-i+1}$, where E_{m-i+1} represents the edges in the graph $\tilde{\mathcal{G}}_{m-i+1}$. By definition, C_{m-i+1} and $B_{v_i} \cap M^{(i)}$ both contain vertex v_i; we now show that these two sets coincide for vertices not equal to v_i.

Choose any $v \in C_{m-i+1}$, $v \neq v_i$. By Proposition 3.11, $\{v, v_i\} \in E_{m-i+1}$ if and only if there exists a path $[v = u_1, u_2, \ldots, u_k, u_{k+1} = v_i]$ in $\mathcal{G}^{\sim}_{\preceq}$ such that $\alpha^{-1}(u_j) < \min(\beta^{-1}(v), \beta^{-1}(v_i), m - i + 2)$ for $2 \leq j \leq k$. This implies that $\alpha^{-1}(u_j) \leq m - i + 1 = \alpha^{-1}(v_i)$, and since u_j and v_i must be distinct, $\alpha^{-1}(u_j) < \alpha^{-1}(v_i)$. Using Lemma B.2, we then have $v \in B_{v_i}$. In addition, the preceding discussion shows that $C_{m-i+1} \subseteq M^{(i)}$, implying $v \in M^{(i)}$. Therefore, $v \in B_{v_i} \cap M^{(i)}$ and $C_{m-i+1} \subset B_{v_i} \cap M^{(i)}$.

Now choose any $v \in B_{v_i} \cap M^{(i)}$, $v \neq v_i$. Then, $v \in B_{v_i}$, in which case Lemma B.2 indicates that there exists a path $[v = u_1, u_2, \ldots, u_k, u_{k+1} = v_i]$ in $\mathcal{G}^{\sim}_{\preceq}$ such that $\alpha^{-1}(u_j) < \alpha^{-1}(v_i) = m - i + 1 < m - i + 2$ for $2 \leq j \leq k$. This also implies $\alpha^{-1}(u_j) < \alpha^{-1}(v_i) \leq \beta^{-1}(v_i)$. Since $v \in M^{(i)}$, either $v \in M$ or $v < v_i$. If $v \in M$, then $\alpha^{-1}(u_j) < \beta^{-1}(v) = \infty$ is always satisfied. If $v \notin M$ and $v < v_i$, then $\alpha^{-1}(v) > \alpha^{-1}(v_i)$, in which case $\alpha^{-1}(u_j) < \alpha^{-1}(v_i) < \alpha^{-1}(v) = \beta^{-1}(v)$. Thus, for $2 \leq j \leq k$, we have $\alpha^{-1}(u_j) < \min(\beta^{-1}(v), \beta^{-1}(v_i), m - i + 2)$, and by Proposition 3.11, $\{v, v_i\} \in E_{m-i+1}$. Therefore, $v \in C_{m-i+1}$, thereby proving $B_{v_i} \cap M^{(i)} \subset C_{m-i+1}$.

(3) Since v_i is a neighborhood separator in $\tilde{\mathcal{G}}_{m-i}(C_{m-i+1})$, we know that the edge $\{v, v_i\}$ is present for all $v \in C_{m-i+1}$ with $v \neq v_i$. Now, choose distinct vertices $v, v' \in C_{m-i+1}$ with $v \neq v_i$ and $v' \neq v_i$. Suppose $\{v, v'\}$ is an edge in $\tilde{\mathcal{G}}_{m-i}(C_{m-i+1})$; then, $\{v, v'\}$ is also an edge in $\tilde{\mathcal{G}}_{m-i}$. By Proposition 3.11, it must be true that the unique path $[v = u_1, u_2, \ldots, u_k, u_{k+1} = v']$ between v and v' in $\mathcal{G}^{\sim}_{\preceq}$ must satisfy $\alpha^{-1}(u_j) < \min(\beta^{-1}(v), \beta^{-1}(v'), m - i + 1)$. Since $m - i + 1 = \alpha^{-1}(v_i)$, this means that $u_j \neq v_i$ for $2 \leq j \leq k$, and therefore, the path between v and v' cannot pass through v_i.

Suppose now that $v, v' \in C_{m-i+1}$ and that the unique path $[v = x_1, x_2, \ldots, x_q, x_{q+1} = v']$ between v and v' in $\mathcal{G}^{\sim}_{\preceq}$ does not pass through v_i. Using the second fact in this lemma, we have $C_{m-i+1} = B_{v_i} \cap M^{(i)}$, and therefore, $v, v' \in B_{v_i}$ and $v, v' \in M^{(i)}$. Since $v \in M^{(i)}$, we must have either $v \in M$ or $v \notin M$ with $v < v_i$, and the same is true for v'. These conditions imply $\alpha^{-1}(v_i) < \beta^{-1}(v)$ and $\alpha^{-1}(v_i) < \beta^{-1}(v')$, or in another form,

$$\alpha^{-1}(v_i) = \min(\beta^{-1}(v), \beta^{-1}(v'), m - i + 1). \tag{B.11}$$

Since $v, v' \in B_{v_i}$, Lemma B.2 indicates that there exist paths $[v = u_1, u_2, \dots, u_k, u_{k+1} = v_i]$ and $[v_i = w_1, w_2, \dots, w_p, w_{p+1} = v']$ in $\mathcal{G}_{\preceq}^{\sim}$ such that $\alpha^{-1}(u_j) < \alpha^{-1}(v_i)$, $2 \leq j \leq k$, and $\alpha^{-1}(w_j) < \alpha^{-1}(v_i)$, $2 \leq j \leq p$. Consider concatenating the path from v to v_i and the path from v_i to v'. This forms a walk $[v, u_2, \dots, u_k, v_i, w_2, \dots, w_p, v']$ (since the unique path from v and v' does not pass through v_i), and the path $[v = x_1, x_2, \dots, x_q, x_{q+1} = v']$ may be formed by removing vertices from this walk. However, the preceding inequalities are also satisfied for this path, *i.e.* $\alpha^{-1}(x_j) < \alpha^{-1}(v_i)$ for $2 \leq j \leq q$. Combining this with (B.11), we get $\alpha^{-1}(x_j) < \alpha^{-1}(v_i) = \min(\beta^{-1}(v), \beta^{-1}(v'), m - i + 1)$. By Proposition 3.11, $\{v, v'\}$ is an edge in the graph $\widetilde{\mathcal{G}}_{m-i}$ and thus an edge in the subgraph $\widetilde{\mathcal{G}}_{m-i}\,(C_{m-i+1})$. ■

Using the preceding two lemmas, we are now in a position to prove Proposition 3.16.

Proposition 3.16 (Marginalization-Invariant Markovianity and Proposition 3.15). *Let $(v_1, \dots, v_m)$ be an ordering on the non-leaf vertices of a rooted tree $\mathcal{G}_{\preceq} = (V, E)$, and let α be any leaf-last ordering on V satisfying (3.50). Assume that $M \subseteq V$ contains all leaf vertices of $\mathcal{G}_{\preceq}$, and let $\mathcal{C}_i$ be defined as in Proposition 3.15. Then, $\mathcal{C}_{m-i+1} = \mathcal{M}_{v_i}$ for $i = 1, \dots, m$.*

Proof. Recall that $\mathcal{M}_{v_i} = \mathcal{S}_{v_i} \cap (B_{v_i} \cap M^{(i)})$ for $i = 1, \dots, m$, and consequently, $\mathcal{M}_{v_i}$ is a collection of vertices from the set $B_{v_i} \cap M^{(i)}$. Similarly, $\mathcal{C}_{m-i+1}$ is a collection of vertices from the set C_{m-i+1}, since $\mathcal{C}_{m-i+1}$ contains the maximal cliques of a subgraph induced by C_{m-i+1}. By the second part of Lemma B.3, $C_{m-i+1} = B_{v_i} \cap M^{(i)}$ for $i = 1, \dots, m$, and hence, $\mathcal{M}_{v_i}$ and $\mathcal{C}_{m-i+1}$ are collections on the same set of vertices. We now show that these two collections are identical.

Choose some $i = 1, \dots, m$. Let the collection $\mathcal{C}_{m-i+1}$, containing the maximal cliques of $\widetilde{\mathcal{G}}_{m-i}\,(C_{m-i+1})$, be represented as $\mathcal{C}_{m-i+1} = \{K_1, K_2, \dots, K_l\}$, and choose any $K_j \in \mathcal{C}_{m-i+1}$. By the third part of Lemma B.3, $v_i \in K_j$ since $\{v, v_i\}$ is an edge in $\widetilde{\mathcal{G}}_{m-i}\,(C_{m-i+1})$ for all $v \neq v_i$ and $v \in C_{m-i+1}$. Now, choose some $v \in K_j$ with $v \neq v_i$. By the third part of Lemma B.3, $\{v, v'\}$ is an edge in $\widetilde{\mathcal{G}}_{m-i}\,(C_{m-i+1})$ if and only if the unique path between v and v' in $\mathcal{G}_{\preceq}^{\sim}$ does not pass through v_i. This is equivalent to saying that v and v' lie in the same subtree of $\mathcal{G}_{\preceq}^{\sim}$ separated by v_i. Therefore, K_j contains v_i plus all vertices in $C_{m-i+1} = B_{v_i} \cap M^{(i)}$ which lie in the same subtree of $\mathcal{G}_{\preceq}^{\sim}$ separated by v_i, and by the definition of $\mathcal{M}_{v_i}$, we must have $K_j \in \mathcal{M}_{v_i}$. We can perform this same argument for each $1 \leq j \leq l$, and since $\mathcal{M}_{v_i}$ and $\mathcal{C}_{m-i+1}$ are defined on the same set of vertices, we must have $\mathcal{M}_{v_i} = \mathcal{C}_{m-i+1}$. ■

■ B.9 Proof of Proposition 3.18

In order to prove Proposition 3.18, we begin with an important lemma which relates the two different sets of maximal cliques $\mathcal{C}_i$ discussed in Propositions 3.16 and 3.18. To begin, assume that an ordering $(v_1, \dots, v_m)$ has been specified on the non-leaf vertices of the graph $\mathcal{G}_{\preceq} = (V, E)$, and for a given set $M \subseteq V$, let the augmented graph $\mathcal{G}_{\preceq}^{\sharp}$ be defined as in Section 3.3.3. Consider any leaf-last ordering α on V which satisfies (3.50), and let $\alpha^{\sharp}$ be the corresponding target-last ordering satisfying (3.51). By saying that $\alpha^{\sharp}$ is the corresponding ordering, we mean that the two orderings satisfy $\alpha(m - i + 1) = v_i$ and $\alpha^{\sharp}(m - i + 1) = v_i^{(d)}$ where $v_i^{(d)}$ is the design vertex associated with the non-leaf vertex v_i. It is important to note that any leaf-last ordering α always has such a corresponding target-last ordering $\alpha^{\sharp}$, and conversely, any target-last ordering $\alpha^{\sharp}$ has a corresponding leaf-last ordering α.

To ease the notational burden in the following lemma, we let $\mathcal{C}_i$ denote the set of maximal cliques considered in Proposition 3.16 for the ordering α, and we now let $\mathcal{C}_i^{\sharp}$ denote the set of maximal cliques considered in Proposition 3.18 for the corresponding ordering $\alpha^{\sharp}$. As the following lemma demonstrates, there exists a simple rule which can be applied to the maximal cliques $\mathcal{C}_i$, $i = 1, \ldots, m$, in order to generate the maximal cliques $\mathcal{C}_i^{\sharp}$.

Lemma B.4 (Maximal Cliques and Augmented Graphs).
Let M, α, $\alpha^{\sharp}$, $\mathcal{C}_i$, and $\mathcal{C}_i^{\sharp}$ be defined as above. Then, $\mathcal{C}_i^{\sharp}$, $i = 1, \ldots, m$, may be obtained from $\mathcal{C}_i$ as follows:

(0) Set $\mathcal{C}_i^{\sharp} = \mathcal{C}_i$ for $i = 1, \ldots, m$.

(1) For each $i = 1, \ldots, m$ and each $v \in \cup\mathcal{C}_i^{\sharp}$, do the following:

(i) If v is a leaf vertex or if v is a non-leaf vertex with $\alpha^{-1}(v) > i$, replace v with $v^{(t)}$.

(ii) If v is a non-leaf vertex with $\alpha^{-1}(v) \leq i$ and $v \in M$, replace v with the tuple $v^{(d)}, v^{(t)}$.

(iii) If v is a non-leaf vertex with $\alpha^{-1}(v) \leq i$ and $v \notin M$, replace v with $v^{(d)}$.

Proof. This result follows directly from the structure of the augmented graph $\mathcal{G}^{\sharp}_{\preceq}$ and the sequence of modified elimination graphs $\widetilde{\mathcal{G}}_i^{\downarrow}$, $i = 1 \ldots, m$ which define the sets of maximal cliques $\mathcal{C}_i^{\sharp}$. First of all, if v is a leaf vertex, then v must be replaced by the target vertex $v^{(t)}$ by the definition of $\mathcal{G}^{\sharp}_{\preceq}$. Namely, all leaf vertices are excluded from being considered design vertices. The remaining steps in this transformation are due to the connectivity of the graph $\mathcal{G}^{\sharp}_{\preceq}$. To see this, notice that $v^{(d)}$ and $v^{(t)}$ have an identical set of neighbors in the graph $\mathcal{G}^{\sharp}_{\preceq}$, and they also have the same neighbors as v does in the graph $\mathcal{G}_{\preceq}$.[1] The vertices $v^{(d)}$ and $v^{(t)}$ also have the same set of neighbors in each of the elimination graphs $\widetilde{\mathcal{G}}_j^{\downarrow}$ up until the point where vertex $v^{(d)}$ is eliminated. Once vertex $v^{(d)}$ is eliminated, vertex $v^{(t)}$ appears alone.

The second part of (i) then follows from the fact that v may be replaced with $v^{(t)}$ once $v^{(d)}$ is no longer a vertex in the elimination graph, *i.e.* $\alpha^{-1}(v) > i$. Item (ii) follows from the fact that both vertices $v^{(d)}$ and $v^{(t)}$ appear in the same maximal cliques before $v^{(d)}$ is eliminated, *i.e.* $\alpha^{-1}(v) \leq i$, assuming that $v \in M$ in which case there is a target vertex $v^{(t)}$. Item (iii) reflects the same idea as (ii), only in this case there is no target vertex $v^{(t)}$ since $v \notin M$. ■

Proposition 3.18 (Marginalization-Invariant Markovianity and Proposition 3.17).
Let $(v_1, \ldots, v_m)$ be an ordering on the non-leaf vertices of a rooted tree $\mathcal{G}_{\preceq} = (V, E)$, and let α be any target-last ordering on $V^{\sharp}$ satisfying (3.51). If $\mathcal{C}_i$ is defined as in Proposition 3.17, then, $\mathcal{C}_{m-i+1} = \mathcal{M}_{v_i}^{\sharp}$ for $i = 1, \ldots, m$.

Proof. This follows directly from Proposition 3.16, the augmentation rule described in Section 2.7.3, and Lemma B.4. Specifically, Proposition 3.16 proves that the maximal cliques $\mathcal{C}_{m-i+1}$ are equal to

[1] Here we mean that the underlying vertices are the same. Namely, the neighbors of vertex v may be $u_1, u_2, \ldots, u_n$ in the graph $\mathcal{G}_{\preceq}$, while the neighbors of $v^{(d)}$ and $v^{(t)}$ are $u_1^{(d)}, u_1^{(t)}, u_2^{(d)}, u_2^{(t)}, \ldots, u_n^{(d)}, u_n^{(t)}$ in the graph $\mathcal{G}^{\sharp}_{\preceq}$.

the families $\mathcal{M}_{v_i}$ when α is a leaf-last ordering on the vertices of $\mathcal{G}_{\preceq}$. If we consider the corresponding ordering $\alpha^\sharp$ on $\mathcal{G}^\sharp_{\preceq}$, then the transformation described in Lemma B.4 may be applied to the families $\mathcal{C}_i$ to give the families $\mathcal{C}^\sharp_i$, which are identical to the families $\mathcal{C}_i$ considered in this proposition. Finally, notice that the transformation in Lemma B.4 is equivalent to the transformation applied to the families $\mathcal{M}_{v_i}$ by the augmentation rule. Since the families $\mathcal{M}^\sharp_{v_i}$ represent the result of applying the augmentation rule to the families $\mathcal{M}_{v_i}$, the result directly follows. ∎

B.10 Proof of Propositions 3.19 and 3.20

Proposition 3.19 (Additivity of Projections).
Let $\mathcal{H} = (V, E)$ be any graph defined on a vertex set V, and let $\mathcal{G} = (V, E')$ be any triangulated supergraph of $\mathcal{H}$, i.e. *$E' \supseteq E$. Suppose $p(x_V)$ and $q(x_V)$ are two densities indexed by V and defined on the same space $\mathcal{X} = \prod_{v \in V} \mathcal{X}_v$. If $q(x)$ factors according to $\mathcal{H}$, then the following decomposition holds,*

$$D\left(p(x)\|q(x)\right) = D\left(p(x)\|p_{\mathcal{G}}(x)\right) + D\left(p_{\mathcal{G}}(x)\|q(x)\right).$$

Proof. Let $\mathcal{C}$ denote the set of maximal cliques in the triangulated graph $\mathcal{G}$. Since $p_{\mathcal{G}}(x)$ factors according to $\mathcal{G}$, we can write

$$p_{\mathcal{G}}(x) = \prod_{C \in \mathcal{C}} \psi_C(x_C), \tag{B.12}$$

for some choice of compatibility functions ψ_C. Since $q(x)$ factors according to $\mathcal{H}$, it also factors according to the supergraph $\mathcal{G}$, and for some choice of $\tilde{\psi}_C$, it can be written as

$$q(x) = \prod_{C \in \mathcal{C}} \tilde{\psi}_C(x_C). \tag{B.13}$$

Notice that $D\left(p(x)\|q(x)\right)$ may be decomposed as follows,

$$\begin{aligned} D\left(p(x)\|q(x)\right) &= \int p(x) \log\left(\frac{p(x)}{q(x)}\right) dx = \int p(x) \log\left(\frac{p(x)}{p_{\mathcal{G}}(x)} \frac{p_{\mathcal{G}}(x)}{q(x)}\right) dx \\ &= \int p(x) \log\left(\frac{p(x)}{p_{\mathcal{G}}(x)}\right) dx + \int p(x) \log\left(\frac{p_{\mathcal{G}}(x)}{q(x)}\right) dx \\ &= D\left(p(x)\|p_{\mathcal{G}}(x)\right) + \int p(x) \log\left(\frac{p_{\mathcal{G}}(x)}{q(x)}\right) dx. \end{aligned}$$

To prove the result, we must show that the last term is equal to $D\left(p_{\mathcal{G}}(x)\|q(x)\right)$. To do this, we use the decompositions for p and q in (B.12) and (B.13),

$$\begin{aligned} \int p(x) \log\left(\frac{p_{\mathcal{G}}(x)}{q(x)}\right) dx &= \int p(x) \log\left(\frac{\prod_{C \in \mathcal{C}} \psi_C(x_C)}{\prod_{C \in \mathcal{C}} \tilde{\psi}_C(x_C)}\right) dx \\ &= \sum_{C \in \mathcal{C}} \left[\int p(x) \log\left(\frac{\psi_C(x_C)}{\tilde{\psi}_C(x_C)}\right) dx\right] \\ &= \sum_{C \in \mathcal{C}} \left[\int p(x_C) \log\left(\frac{\psi_C(x_C)}{\tilde{\psi}_C(x_C)}\right) dx_C\right]. \end{aligned} \tag{B.14}$$

In the last equality, we have integrated out the variables x_{V-C}.

Now, compare the decomposition obtained in (B.14) to the following decomposition,

$$\begin{aligned} D\left(p_{\mathcal{G}}(x)\|q(x)\right) &= \int p_{\mathcal{G}}(x)\log\left(\frac{p_{\mathcal{G}}(x)}{q(x)}\right)dx \\ &= \sum_{C\in\mathcal{C}}\left[\int p_{\mathcal{G}}(x_C)\log\left(\frac{\psi_C(x_C)}{\tilde{\psi}_C(x_C)}\right)dx_C\right]. \end{aligned} \tag{B.15}$$

This decomposition was derived in the same manner used to derive (B.14), and as such, we have omitted the intermediate steps. By the definition of the projection $p_{\mathcal{G}}(x)$, we know that $p_{\mathcal{G}}(x_C) = p(x_C)$ for all $C \in \mathcal{C}$. Therefore, the sum of terms in (B.14) and (B.15) is identical, thereby proving the result. ■

Proposition 3.20 (The Geometry of the Mapping $\mathcal{F}^M$).
Let $p \in \mathcal{P}^M(V,d)$ and $q \in \mathcal{P}(V,d)$. Then, the following decomposition holds,

$$D\left(p(x)\|q(x)\right) = D\left(p(x)\|\mathcal{F}^M(q(x))\right) + D\left(\mathcal{F}^M(q(x))\|q(x)\right).$$

Proof. Defining $\bar{p}(x) \triangleq \mathcal{F}^M(q(x))$ and manipulating the integrals in the definition of KL, we get

$$\begin{aligned} D\left(p(x)\|q(x)\right) = \int p(x)\log\left(\frac{p(x)}{\bar{p}(x)}\frac{\bar{p}(x)}{q(x)}\right)dx &= \int p(x)\log\left(\frac{p(x)}{\bar{p}(x)}\right)dx + \int p(x)\log\left(\frac{\bar{p}(x)}{q(x)}\right)dx \\ &= D\left(p(x)\|\bar{p}(x)\right) + \int p(x)\log\left(\frac{\bar{p}(x)}{q(x)}\right)dx. \end{aligned}$$

We now show that the final term in the preceding equation is equal to $D\left(\bar{p}(x)\|q(x)\right)$. Using the fact that $\bar{p}(x) = q(x_{V-M}|x_M)p^*(x_M)$ and integrating out the variables x_{V-M} gives

$$\begin{aligned} \int p(x)\log\left(\frac{\bar{p}(x)}{q(x)}\right)dx &= \int p(x)\log\left(\frac{p^*(x_M)}{q(x_M)}\right)dx \\ &= \int p^*(x_M)\log\left(\frac{p^*(x_M)}{q(x_M)}\right)dx_M = D\left(p^*(x_M)\|q(x_M)\right). \end{aligned}$$

Notice that the last line in the preceding decomposition follows from the fact that $p(x_M) = p^*(x_M)$, and therefore, we must require $p \in \mathcal{P}^M(V,d)$ for this relationship to hold. Finally, using (3.57), we have $D\left(p^*(x_M)\|q(x_M)\right) = D\left(\bar{p}(x)\|q(x)\right)$. ■

■ B.11 Proof of Proposition 3.22

Proposition 3.22 (Relationship Between Solutions to $\widetilde{\mathcal{P}}_{\mathcal{G}}^M$ and $\widetilde{\mathcal{Q}}$).
Let $\mathcal{G}_{\preceq}$ be a rooted tree defined on vertex set V, and let $p^*(x_M)$ be a given target density. If a graph $\mathcal{G} = (V,E)$ is a supergraph of $\mathcal{G}_{\preceq}^{\sim}$ and has a clique equal to M, the mapping $p \longrightarrow p^T$ is a surjection from the solution set of problem $\widetilde{\mathcal{P}}_{\mathcal{G}}^M$ onto the solution set of problem $\widetilde{\mathcal{Q}}$.

Proof. Let $\hat{p}$ be any solution to problem $\widetilde{\mathcal{P}}_{\mathcal{G}}^M$. We first show that $\hat{p}^T$ is a solution to problem $\widetilde{\mathcal{Q}}$. To do this, suppose $\hat{p}^T$ is not a solution to problem $\widetilde{\mathcal{Q}}$, and let $\hat{q}$ be any solution to problem $\widetilde{\mathcal{Q}}$. By

Proposition 3.21, the density $\bar{p} \triangleq \mathcal{F}^M(\hat{q})$ satisfies $\hat{q} = \bar{p}^T$, and furthermore, $\bar{p} \in \mathcal{P}_\mathcal{G}^M(V, d)$ for all graphs $\mathcal{G}$ which are supergraphs of $\mathcal{G}_{\precsim}$ and have a clique equal to M. Therefore, we have a density $\bar{p} \in \mathcal{P}_\mathcal{G}^M(V, d)$ such that

$$D\left(p^*(x_M) \| \bar{p}^T(x_M)\right) = D\left(p^*(x_M) \| \hat{q}(x_M)\right) < D\left(p^*(x_M) \| \hat{p}^T(x_M)\right),$$

but this contradicts the fact that $\hat{p}$ is a solution to problem $\widetilde{\mathcal{P}}_\mathcal{G}^M$. Therefore, $\hat{p}^T$ is a solution to problem $\widetilde{\mathcal{Q}}$.

We now show that $\mathcal{F}^M(\cdot)$ is a mapping from the solution set of $\widetilde{\mathcal{Q}}$ to the solution set of $\widetilde{\mathcal{P}}_\mathcal{G}^M$. Let $\hat{q}$ be any solution to problem $\widetilde{\mathcal{Q}}$, and define $\bar{p} \triangleq \mathcal{F}^M(\hat{q})$. We have $\bar{p} \in \mathcal{P}_\mathcal{G}^M(V, d)$ as long as $\mathcal{G}$ is a supergraph of $\mathcal{G}_{\precsim}$ and has a clique equal to M. Suppose that $\bar{p}$ is not a solution to problem $\widetilde{\mathcal{P}}_\mathcal{G}^M$, so that there exists a $\hat{p}$ such that

$$D\left(p^*(x_M) \| \hat{p}^T(x_M)\right) < D\left(p^*(x_M) \| \bar{p}^T(x_M)\right) = D\left(p^*(x_M) \| \hat{q}(x_M)\right)$$

and where the final equality follows from Proposition 3.21. This inequality however contradicts the fact that $\hat{q}$ is a solution to problem $\widetilde{\mathcal{Q}}$, and therefore, we must have that $\bar{p}$ is a solution to problem $\widetilde{\mathcal{P}}_\mathcal{G}^M$.

We have shown that $p \longrightarrow p^T$ is a mapping from the solution set of $\widetilde{\mathcal{P}}_\mathcal{G}^M$ to the solution set of $\widetilde{\mathcal{Q}}$ and $\mathcal{F}^M(\cdot)$ is a mapping from the solution set of $\widetilde{\mathcal{Q}}$ to the solution set of $\widetilde{\mathcal{P}}_\mathcal{G}^M$. Further, Proposition 3.21 indicates that $p \longrightarrow p^T$ is the inverse of $\mathcal{F}^M(\cdot)$ on the domain equal to the solution set of $\widetilde{\mathcal{Q}}$. Hence, $p \longrightarrow p^T$ is a surjection. ■

Appendix C

Proofs for Chapter 4

C.1 Proof of Proposition 4.1

Proposition 4.1 (An Invariance Property of Solutions to Problem $\widetilde{\mathcal{Q}}(\Theta)$).
Let $\mathcal{G}_{\preceq}$ be a rooted tree defined on vertex set V, and let $p^*(x_M)$ be a given target density. Let Θ and Γ be specified sets which index the densities $\{q(x|\theta)\}_{\theta\in\Theta} \subset \mathcal{P}_{\mathcal{G}_{\preceq}}(V,d)$ and $\{p(x|\gamma)\}_{\gamma\in\Gamma} \subset \mathcal{P}^M(V,d)$ respectively. If the mapping $\mathcal{T}(\cdot)$ in (4.21) and the mapping $\mathcal{M}(\cdot)$ in (4.24) both exist and if $\hat{\theta}$ is a solution to problem $\widetilde{\mathcal{Q}}(\Theta)$, then $\hat{\theta} = \mathcal{T}\left(\mathcal{M}\left(\hat{\theta}\right)\right)$.

Proof. Define $\hat{\gamma} \triangleq \mathcal{M}(\hat{\theta})$ and $\bar{\theta} \triangleq \mathcal{T}(\hat{\gamma})$. We want to show that $\bar{\theta} = \hat{\theta}$. Using the decomposition in (4.1), we can write

$$\begin{aligned} D\left(p^*(x_M)\|q(x_M|\hat{\theta})\right) &= D\left(p(x|\hat{\gamma})\|q(x|\hat{\theta})\right) - D\left(p(x|\hat{\gamma})\|\mathcal{F}^M(q(x|\hat{\theta}))\right) \\ &= D\left(p(x|\hat{\gamma})\|q(x|\hat{\theta})\right), \end{aligned}$$

where the last equality follows from the definition of $\hat{\gamma} = \mathcal{M}(\hat{\theta})$.

Next, the following inequality must be satisfied, since $\bar{\theta} = \mathcal{T}(\hat{\gamma})$ is by definition the unique minimizer of $D\left(p(x|\hat{\gamma})\|q(x|\theta)\right)$ over all $\theta \in \Theta$,

$$D\left(p^*(x_M)\|q(x_M|\hat{\theta})\right) = D\left(p(x|\hat{\gamma})\|q(x|\hat{\theta})\right) > D\left(p(x|\hat{\gamma})\|q(x|\bar{\theta})\right).$$

Finally, the decomposition in (4.1) is used once again to give

$$\begin{aligned} D\left(p^*(x_M)\|q(x_M|\hat{\theta})\right) &> D\left(p(x|\hat{\gamma})\|q(x|\bar{\theta})\right) \\ &= D\left(p(x|\hat{\gamma})\|\mathcal{F}^M(q(x|\bar{\theta}))\right) + D\left(p^*(x_M)\|q(x_M|\bar{\theta})\right). \end{aligned} \tag{C.1}$$

The inequality in (C.1) contradicts the fact that $\hat{\theta}$ is a solution to problem $\widetilde{\mathcal{Q}}(\Theta)$. Since no such $\bar{\theta}$ can exist, we must have $\bar{\theta} = \hat{\theta}$, in which case the inequality in (C.1) becomes an equality and the divergence $D\left(p(x|\hat{\gamma})\|\mathcal{F}^M(q(x|\bar{\theta}))\right) = D\left(p(x|\hat{\gamma})\|\mathcal{F}^M(q(x|\hat{\theta}))\right)$ is zero by the definition of $\hat{\gamma}$. ■

C.2 Proof of Proposition 4.2

Proposition 4.2 (Relationship Between Solutions to $\widetilde{\mathcal{P}}^M(\Gamma)$ and $\widetilde{\mathcal{Q}}(\Theta)$).
Suppose the assumptions stated in Proposition 4.1 hold; in particular, suppose the mappings $\mathcal{T}(\cdot)$ and $\mathcal{M}(\cdot)$ exist. Then, the mapping $\mathcal{T}(\cdot)$ is a surjection from the solution set of problem $\widetilde{\mathcal{P}}^M(\Gamma)$ onto the solution set of problem $\widetilde{\mathcal{Q}}(\Theta)$.

Proof. Let $\hat{\gamma}$ be any solution to problem $\widetilde{\mathcal{P}}^M(\Gamma)$. We first show that $\hat{\theta} \triangleq \mathcal{T}(\hat{\gamma})$ is a solution to problem $\widetilde{\mathcal{Q}}(\Theta)$. To do this, suppose $\hat{\theta}$ is not a solution to problem $\widetilde{\mathcal{Q}}(\Theta)$, and let $\bar{\theta}$ be any solution to problem $\widetilde{\mathcal{Q}}(\Theta)$. Define $\bar{\gamma} \triangleq \mathcal{M}(\bar{\theta})$. By Proposition 4.1, since $\bar{\theta}$ is a solution to problem $\widetilde{\mathcal{Q}}(\Theta)$, it satisfies $\bar{\theta} = \mathcal{T}(\bar{\gamma})$. Since $\hat{\theta}$ is not a solution to problem $\widetilde{\mathcal{Q}}(\Theta)$, we must then have

$$\begin{aligned} D\left(p^*(x_M) \| q(x_M|\mathcal{T}(\bar{\gamma}))\right) &= D\left(p^*(x_M) \| q(x_M|\bar{\theta})\right) \\ &< D\left(p^*(x_M) \| q(x_M|\hat{\theta})\right) = D\left(p^*(x_M) \| q(x_M|\mathcal{T}(\hat{\gamma}))\right). \end{aligned}$$

This inequality contradicts the fact that $\hat{\gamma}$ is a solution to problem $\widetilde{\mathcal{P}}^M(\Gamma)$, and we must therefore have that $\hat{\theta} = \mathcal{T}(\hat{\gamma})$ is a solution to problem $\widetilde{\mathcal{Q}}(\Theta)$.

We now show that $\mathcal{M}(\cdot)$ is a mapping from the solution set of $\widetilde{\mathcal{Q}}(\Theta)$ to the solution set of $\widetilde{\mathcal{P}}^M(\Gamma)$. Let $\hat{\theta}$ be any solution to problem $\widetilde{\mathcal{Q}}(\Theta)$, and define $\hat{\gamma} \triangleq \mathcal{M}(\hat{\theta})$. By Proposition 4.1, $\hat{\theta}$ must satisfy $\hat{\theta} = \mathcal{T}(\hat{\gamma})$. Suppose that $\hat{\gamma}$ is not a solution to problem $\widetilde{\mathcal{P}}^M(\Gamma)$, so that there exists a $\bar{\gamma}$ and a $\bar{\theta} \triangleq \mathcal{T}(\bar{\gamma})$ such that

$$\begin{aligned} D\left(p^*(x_M) \| q(x_M|\bar{\theta})\right) &= D\left(p^*(x_M) \| q(x_M|\mathcal{T}(\bar{\gamma}))\right) \\ &< D\left(p^*(x_M) \| q(x_M|\mathcal{T}(\hat{\gamma}))\right) = D\left(p^*(x_M) \| q(x_M|\hat{\theta})\right) \end{aligned}$$

This inequality contradicts the fact that $\hat{\theta}$ is a solution to problem $\widetilde{\mathcal{Q}}(\Theta)$, and therefore, we must have that $\hat{\gamma}$ is a solution to problem $\widetilde{\mathcal{P}}^M(\Gamma)$.

We have shown that $\mathcal{T}(\cdot)$ is a mapping from the solution set of $\widetilde{\mathcal{P}}^M(\Gamma)$ to the solution set of $\widetilde{\mathcal{Q}}(\Theta)$ and that $\mathcal{M}(\cdot)$ is a mapping from the solution set of $\widetilde{\mathcal{Q}}(\Theta)$ to the solution set of $\widetilde{\mathcal{P}}^M(\Gamma)$. Furthermore, Proposition 4.1 indicates that $\mathcal{T}(\cdot)$ is the inverse of $\mathcal{M}(\cdot)$ on the domain equal to the solution set of $\widetilde{\mathcal{Q}}(\Theta)$. Hence, the mapping $\mathcal{T}(\cdot)$ is the needed surjection. ■

■ C.3 Application of Algorithm 4.2 to Gaussian Multiscale Densities

In Section 4.4.2, we present a specific algorithm for calculating the matrix quantities necessary to perform the expectation step of the EM algorithm. In this section, we show how these quantities are derived. Specifically, we prove the equalities in (4.46), (4.47), and (4.48). For simplicity of notation, we no longer denote the fact that each density is parameterized by the set θ.

The first equality in (4.46), which corresponds to the merge step of the algorithm, is derived using the product of densities previously provided in (4.18) and included here for reference,

$$q(x_t|x_{L_t}) = \left[\frac{\prod_{s\in\chi(t)} q(x_{L_s})}{q(x_{L_t})}\right] \cdot \left[\frac{1}{q(x_t)^{|\chi(t)|-1}}\right] \cdot \prod_{s\in\chi(t)} q(x_t|x_{L_s}). \tag{C.2}$$

Notice that the first term in (C.2) is constant with respect to x_t. Using this fact, as well as the parametrization provided in (4.44), the Gaussian density $q(x_t|x_{L_t})$ can be written as follows,

$$\begin{aligned} q(x_t|x_{L_t}) &= \alpha_0 \times \\ &\exp\left[-\frac{1}{2}\left(1 - |\chi(t)|\right) x_t^T Q_t^{-1} x_t - \frac{1}{2}\sum_{s\in\chi(t)} \left(x_t - M_s^p x_{L_s}\right)^T \left[R_s^p\right]^{-1} \left(x_t - M_s^p x_{L_s}\right)\right], \end{aligned} \tag{C.3}$$

where α_0 is constant with respect to x_t.

By defining the matrix R_t^m as follows,

$$R_t^m = \left[(1 - |\chi(t)|)\, Q_t^{-1} + \sum_{s \in \chi(t)} [R_s^p]^{-1} \right]^{-1}, \tag{C.4}$$

the expression in (C.3) can then be manipulated into the form of a Gaussian density,

$$\begin{aligned} &q(x_t|x_{L_t}) = \alpha_0 \alpha_1 \times \\ &\exp\left[-\frac{1}{2} \left(x_t - R_t^m \sum_{s \in \chi(t)} [R_s^p]^{-1} M_s^p x_{L_s} \right)^T [R_t^m]^{-1} \left(x_t - R_t^m \sum_{s \in \chi(t)} [R_s^p]^{-1} M_s^p x_{L_s} \right) \right]. \end{aligned} \tag{C.5}$$

The argument to $\exp(\cdot)$ in the preceding expression contains the terms included in (C.3), plus an additional cross term which only involves x_{L_t}. This additional term is accounted for by the scale factor α_1.

Since the density $q(x_t|x_{L_t})$ in (C.5) has the form of a Gaussian, we can immediately write down its mean and covariance. Specifically, its covariance is specified by the matrix R_t^m defined in (C.4), and its mean (as a function of x_{L_t}) is given by the following,

$$\mu(x_t|x_{L_t}) = R_t^m \sum_{s \in \chi(t)} [R_s^p]^{-1} M_s^p x_{L_s}.$$

Using the fact that the mean of $q(x_t|x_{L_t})$ is defined in (4.44b) to be of the form $\mu(x_t|x_{L_t}) = M_t^m x_{L_t}$, the matrix M_t^m can be written as follows,

$$M_t^m = R_t^m \left[\ [R_{s_1}^p]^{-1} M_{s_1}^p \ \middle|\ [R_{s_2}^p]^{-1} M_{s_2}^p \ \middle|\ \cdots \ \middle|\ [R_{s_{q_t}}^p]^{-1} M_{s_{q_t}}^p \ \right], \tag{C.6}$$

assuming that $x_{L_t} \triangleq \left[\ x_{L_{s_1}} \ \middle|\ x_{L_{s_2}} \ \middle|\ \cdots \ \middle|\ x_{L_{s_{q_t}}} \ \right]^T$. The expressions for R_t^m and M_t^m in (C.4) and (C.6) agree with those previously given in (4.46a) and (4.46b) respectively.

To prove the two remaining equalities, we first derive an intermediate result. Consider a Gaussian density $q(x, y|z)$ which factors as $q(x, y|z) = q(x|y)q(y|z)$, and suppose these densities are parameterized as follows,

$$q(x, y|z) = N\left(x, y; Az, Q\right), \tag{C.7a}$$

$$q(x|y) = N\left(x; A_y y, Q_y\right), \tag{C.7b}$$

$$q(y|z) = N\left(y; A_z z, Q_z\right). \tag{C.7c}$$

Our goal is to determine A and Q as a function of the parameters A_y, A_z, Q_y, and Q_z.

Using the parametrization in (C.7), we can write the following for the density $q(x, y|z)$,

$$\begin{aligned} q(x, y|z) &= q(x|y)q(y|z) \\ &= \alpha \exp\left[-\frac{1}{2}(x - A_y y)^T Q_y^{-1} (x - A_y y) - \frac{1}{2}(y - A_z z)^T Q_z^{-1} (y - A_z z) \right]. \end{aligned}$$

The preceding equation can be manipulated into the form of a Gaussian density by defining

$$Q^{-1} = \begin{bmatrix} Q_y^{-1} & -Q_y^{-1}A_y \\ -A_y^T Q_y^{-1} & Q_z^{-1} + A_y^T Q_y^{-1} A_y \end{bmatrix}, \tag{C.8}$$

so that we then have

$$q(x, y|z) = \alpha \exp\left[-\frac{1}{2} \left(\begin{bmatrix} x \\ y \end{bmatrix} - Q \begin{bmatrix} 0 \\ Q_z^{-1} A_z z \end{bmatrix} \right)^T Q^{-1} \left(\begin{bmatrix} x \\ y \end{bmatrix} - Q \begin{bmatrix} 0 \\ Q_z^{-1} A_z z \end{bmatrix} \right) \right]. \tag{C.9}$$

The block-partitioned matrix Q^{-1} in (C.8) can be inverted using the Schur complement to give the following,

$$Q = \begin{bmatrix} Q_y + A_y Q_z A_y^T & A_y Q_z \\ Q_z A_y^T & Q_z \end{bmatrix}. \tag{C.10}$$

Using this expression for Q in (C.9) then gives the desired result,

$$q(x, y|z) = \alpha \exp\left[-\frac{1}{2} \left(\begin{bmatrix} x \\ y \end{bmatrix} - \begin{bmatrix} A_y A_z \\ A_z \end{bmatrix} z \right)^T Q^{-1} \left(\begin{bmatrix} x \\ y \end{bmatrix} - \begin{bmatrix} A_y A_z \\ A_z \end{bmatrix} z \right) \right]. \tag{C.11}$$

Therefore, $q(x, y|z)$ is Gaussian with mean Az where

$$A = \begin{bmatrix} A_y A_z \\ A_z \end{bmatrix}$$

and with covariance Q as specified in (C.10).

For our purposes, we are interested in two important densities derived from the conditional density $q(x, y|z)$. The first is the density $q(x|z)$ obtained by marginalizing $q(x, y|z)$ over y. Using (C.11), the parameters of $q(x|z)$ are easily identified by looking at the correct matrix sub-blocks, *i.e.* $q(x|z)$ has mean $A_y A_z z$ and covariance $Q_y + A_y Q_z A_y^T$. The second density of interest is the conditional density $q(y|x, z)$. Using the standard formula for computing the parameters of a Gaussian conditional density, the mean $\mu(y|x, z)$ of $q(y|x, z)$ is given by

$$\mu(y|x, z) = A_z z + Q_z A_y^T \left(Q_y + A_y Q_z A_y^T \right)^{-1} (x - A_y A_z z) = \left[\, J \mid (I - J A_y) A_z \, \right] \begin{bmatrix} x \\ z \end{bmatrix},$$

where

$$J \triangleq Q_z A_y^T \left(Q_y + A_y Q_z A_y^T \right)^{-1},$$

and the conditional covariance matrix $Q_{y|x,z}$ of $q(y|x, z)$ is given by

$$Q_{y|x,z} = Q_z - Q_z A_y^T \left(Q_y + A_y Q_z A_y^T \right)^{-1} A_y Q_z = (I - J A_y) Q_z.$$

Using the preceding results, the equalities in (4.47) and (4.48) directly follow. Specifically, if we equate x_s with y, x_t with x, and x_{L_s} with z, then the parameters of the densities $q(x|z)$ and $q(y|x, z)$ can be used to determine the means and covariances of the densities $q(x_t|x_{L_s})$ and $q(x_s|x_t, x_{L_s})$ respectively. Doing so leads to the same matrix equalities previously derived in (4.47) and (4.48).

■ C.4 Proof of Proposition 4.5

Proposition 4.5 (Rescaling Multiscale Models to Have Identity State Covariances). *Let two zero-mean Gaussian multiscale densities $q(x|\theta) = N(x; 0, Q)$ and $q(x|\bar{\theta}) = N(x; 0, \bar{Q})$ be specified such that the marginals $\bar{Q}_v$ and $\bar{Q}_{uv}$ of $\bar{Q}$ are defined in terms of the marginals Q_v and Q_{uv} of Q according to (4.56). Then, $\bar{Q}_v = I$ for all non-leaf vertices v, and the two marginal covariances Q_M and $\bar{Q}_M$ are identical.*

Proof. The fact that $\bar{Q}_v = I$ for all non-leaf vertices v follows by the definition of the mapping in (4.56). Furthermore, by definition $\bar{Q}_v = Q_v$ for all $v \in M$. Therefore, all that remains to be proven is that $\bar{Q}_{ij} = Q_{ij}$ for all $i \neq j$ and $i, j \in M$. To see this, we use the fact that the covariance Q_{ij} may be expressed in terms of the edge covariances Q_{uv} and state covariances Q_v of the multiscale model [38]. Specifically, let $i, j \in M$, and consider the unique path in $\mathcal{G}_{\tilde{\preceq}}$ between i and j. This path has the following form,

$$[u_0 = i, u_1, u_2, \ldots, u_n = t = v_m, \ldots, v_1, v_0 = j], \tag{C.12}$$

where vertex t is a common ancestor of i and j in the rooted tree $\mathcal{G}_{\preceq}$.

Given the path in (C.12), the covariance Q_{ij} may be expressed as follows,

$$Q_{ij} = \left[\prod_{k=1}^{n} Q_{u_{k-1},u_k} Q_{u_k}^{-1} \right] Q_t \left[\prod_{k=1}^{m} Q_{v_{k-1},v_k} Q_{v_k}^{-1} \right]^T . \tag{C.13}$$

The preceding expression along with the mapping in (4.56) then implies the following form for $\bar{Q}_{ij}$,

$$\begin{aligned}
\bar{Q}_{ij} &= \left[\prod_{k=1}^{n} \bar{Q}_{u_{k-1},u_k} \bar{Q}_{u_k}^{-1} \right] \bar{Q}_t \left[\prod_{k=1}^{m} \bar{Q}_{v_{k-1},v_k} \bar{Q}_{v_k}^{-1} \right]^T \\
&= Q_{i,u_1} Q_{u_1}^{-1/2} \left[\prod_{k=2}^{n} Q_{u_{k-1}}^{-1/2} Q_{u_{k-1},u_k} Q_{u_k}^{-1/2} \right] I \left[\prod_{k=2}^{m} Q_{v_{k-1}}^{-1/2} Q_{v_{k-1},v_k} Q_{v_k}^{-1/2} \right]^T Q_{v_1}^{-1/2} Q_{v_1,j} \\
&= \left[\prod_{k=1}^{n} Q_{u_{k-1},u_k} Q_{u_k}^{-1} \right] Q_t \left[\prod_{k=1}^{m} Q_{v_{k-1},v_k} Q_{v_k}^{-1} \right]^T = Q_{ij}.
\end{aligned}$$

Consequently, this proves that $\bar{Q}_M = Q_M$. ■

■ C.5 Proof of Proposition 4.6

Proposition 4.6 (Convergence of the Sequence $\{\bar{\theta}^{(i)}\}$ for the Gaussian Realization Problem). *Let Θ index the set of zero-mean Gaussian multiscale densities with positive-definite covariances, and suppose $\bar{\theta}^{(0)}$ is an initial starting point for the sequence $\{\bar{\theta}^{(i)}\}$ in (4.57) which satisfies $\bar{\varepsilon}_0 = D\left(p^*(x_M) \| q(x_M|\bar{\theta}^{(0)})\right) < \infty$. Then, the sequence $\{\bar{\theta}^{(i)}\}$ converges to a fixed point $\hat{\theta}$ which satisfies $\hat{\theta} = \mathcal{T}(\mathcal{M}(\hat{\theta}))$.*

Proof. Using Proposition 4.3, we need to prove that $\bar{\varepsilon}_i \triangleq D\left(p^*(x_M) \| q(x_M|\bar{\theta}^{(i)})\right)$ is monotonically decreasing and that $\{\bar{\theta}^{(i)}\}$ is a bounded sequence. The latter follows directly from the fact that

$$\bar{\Theta}_0 \triangleq \left\{ \bar{\theta} \in \bar{\Theta} \,\middle|\, D\left(p^*(x_M) \| q(x_M|\bar{\theta})\right) \leq D\left(p^*(x_M) \| q(x_M|\bar{\theta}^{(0)})\right) \right\}$$

is bounded for every initial starting point $\bar{\theta}^{(0)}$ satisfying $\bar{\varepsilon}_0 < \infty$ and from the fact that $\{\bar{\varepsilon}_i\}$ is non-increasing, as we now show.

To prove that $\{\bar{\varepsilon}_i\}$ is non-increasing, we use the previous inequalities derived in (4.28), (4.29), and (4.30),

$$\begin{aligned}
\bar{\varepsilon}_i &= D\left(p(x|\gamma^{(i+1)})\|q(x|\bar{\theta}^{(i)})\right) - D\left(p(x|\gamma^{(i+1)})\|\mathcal{F}^M\left(q(x|\bar{\theta}^{(i)})\right)\right) && \text{(C.14a)}\\
&= D\left(p(x|\gamma^{(i+1)})\|q(x|\bar{\theta}^{(i)})\right) && \text{(C.14b)}\\
&\geq D\left(p(x|\gamma^{(i+1)})\|q(x|\theta^{(i+1)})\right) && \text{(C.14c)}\\
&= D\left(p(x|\gamma^{(i+1)})\|\mathcal{F}^M\left(q(x|\theta^{(i+1)})\right)\right) + D\left(p^*(x_M)\|q(x_M|\theta^{(i+1)})\right) && \text{(C.14d)}\\
&= D\left(p(x|\gamma^{(i+1)})\|\mathcal{F}^M\left(q(x|\theta^{(i+1)})\right)\right) + D\left(p^*(x_M)\|q(x_M|\bar{\theta}^{(i+1)})\right) && \text{(C.14e)}\\
&\geq \bar{\varepsilon}_{i+1}. && \text{(C.14f)}
\end{aligned}$$

Consequently, $\{\bar{\varepsilon}_i\}$ is non-increasing and converges in the limit. Since the mapping $\mathcal{T}(\cdot)$ exists in the Gaussian case, we then have from Proposition 4.3 that $\{\bar{\theta}^{(i)}\}$ converges to a fixed point $\hat{\theta}$.

Using the iterations in (4.57), the fixed point $\hat{\theta}$ must satisfy $\hat{\theta} = \mathcal{R}\left(\mathcal{T}\left(\mathcal{M}\left(\hat{\theta}\right)\right)\right)$. However, at the point of convergence, the inequality in (C.14c) becomes equality, and we have $\hat{\theta} = \mathcal{T}(\mathcal{M}(\hat{\theta}))$. This proves that the iterations in (4.57) converge to a point which lies in the set $\bar{\Theta}$, so that $\mathcal{T}(\mathcal{M}(\hat{\theta}))$ is invariant under the mapping $\mathcal{R}(\cdot)$. ■

Bibliography

[1] S. Amari. Differential geometry of curved exponential families – curvature and information loss. *Annals of Statistics*, 10(2):357–385, June 1982.

[2] S. Amari. Information geometry on hierarchy of probability distributions. *IEEE Transactions on Information Theory*, 47(5):1701–1711, July 2001.

[3] F. R. Bach and M. I. Jordan. Kernel independent component analysis. *The Journal of Machine Learning Research*, 3:1–48, 2003.

[4] O. Barndorff-Nielsen. *Information and Exponential Families.* Wiley, Chichester, 1978.

[5] M. Basseville. Distance measures for signal processing and pattern recognition. *Signal Processing*, 18:349–369, 1989.

[6] C. Berroux, A. Glavieux, and P. Thitmajshima. Near Shannon limit error-correcting coding and decoding. In *Proceedings of ICC*, pages 1064–1070, 1993.

[7] A. Berry, J. R. S. Blair, and P. Heggernes. *Graph Theoretical Concepts in Computer Science*, chapter Maximum Cardinality Search for Computing Minimal Triangulations. Springer-Verlag, 2002. Lecture Notes in Computer Science.

[8] D. P. Bertsekas. *Dynamic Programming and Optimal Control.* Athena Publishing, 2nd edition, November 2000.

[9] D. P. Bertsekas. *Nonlinear Programming.* Athena Publishing, 2nd edition, April 2004.

[10] J. Besag. Spatial interaction and the statistical analysis of lattice systems. *Journal of the Royal Statistical Society, Series B*, pages 192–236, 1974.

[11] C. A. Bouman and M. Shapiro. A multiscale random field model for Bayesian image segmentation. *IEEE Transactions on Image Processing*, 3(2):162–177, March 1994.

[12] N. N. Chentsov. Statistical decision rules and optimal inference. In *Translations of Mathematical Monographs*, volume 53. American Mathematical Society, 1982.

[13] K. Chou. *A stochastic modeling approach to multiscale signal processing.* PhD thesis, Massachusetts Institute of Technology, May 1991.

[14] K. Chou, A. S. Willsky, and A. Benveniste. Multiscale recursive estimation, data fusion, and regularization. *IEEE Transactions on Automatic Control*, 39(3):464–478, March 1994.

[15] K. Chou, A. S. Willsky, and R. Nikoukhah. Multiscale systems, Kalman filters, and Riccati equations. *IEEE Transactions on Automatic Control*, 39(3):479–492, March 1994.

[16] T. Constantinescu. *Schur Parameters, Factorization and Dilation Problems*, volume 82 of *Operator Theory Advances and Applications*. Birkhauser Verlag, 1996.

[17] T. M. Cover and J. A. Thomas. *Elements of Information Theory*. Wiley Series in Telecommunications. John Wiley & Sons, Inc., 1991.

[18] I. Csiszár. I-divergence geometry of probability distributions and minimization problems. *Annals of Probability*, 3(1):146–158, February 1975.

[19] I. Csiszár. A geometric interpretation of Darroch and Ratcliff's generalized iterative scaling. *The Annals of Statistics*, 17(3):1409–1413, September 1989.

[20] I. Csiszár and G. Tusnády. Information geometry and alternating minimization procedures. *Statistics and Decisions*, Supplement Issue 1:205–237, 1984.

[21] M. M. Daniel. *Multiresolution Statistical Modeling with Application to Modeling Groundwater Flow*. PhD thesis, Massachusetts Institute of Technology, February 1997.

[22] M. M. Daniel and A. S. Willsky. *Fractals in Engineering*, chapter Modeling and estimation of fractional Brownian motion using multiresolution stochastic processes, pages 124–137. Springer, 1997.

[23] M. M. Daniel and A. S. Willsky. A multiresolution methodology for signal-level fusion and data assimilation with applications to remote sensing. *Proceedings of the IEEE*, 85(1):164–180, January 1997.

[24] M. M. Daniel and A. S. Willsky. The modeling and estimation of statistically self-similar processes in a multiresolution framework. *IEEE Transactions on Information Theory*, 45(3):955–970, April 1999.

[25] M. M. Daniel, A. S. Willsky, and D. McLaughlin. A multiscale approach for estimating solute travel time distributions. *Advances in Water Resources*, 23:653–665, 2000.

[26] M. H. DeGroot and M. J. Schervish. *Probability and Statistics*. Addison Wesley, 3rd edition, 2001.

[27] A. Dempster, N. Laird, and D. Rubin. Maximum likelihood estimation from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society Series B*, 39, 1977.

[28] B. Efron. The geometry of exponential families. *The Annals of Statistics*, 6(2):362–376, 1978.

[29] P. Fieguth. *Application of multiscale estimation to large scale multidimensional imaging and remote sensing problems*. PhD thesis, Massachusetts Institute of Technology, June 1995.

[30] P. Fieguth, W. Karl, A. S. Willsky, and C. Wunsch. Multiresolution optimal interpolation and statistical analysis of TOPEX/POSEIDON satellite altimetry. *IEEE Transactions on Geoscience and Remote Sensing*, 33(2):280–292, March 1995.

[31] P. Fieguth, D. Menemenlis, T. Ho, A. S. Willsky, and C. Wunsch. Mapping Mediterranean altimeter data with a multiresolutions optimal interpolation algorithm. *Journal of Atmospheric and Oceanic Technology*, 15:535–546, April 1998.

[32] P. Fieguth, D. Menemenlis, C. Wunsch, and A. S. Willsky. Adaptation of a fast optimal interpolation algorithm to the mapping of oceanographic data. *Journal of Geophysical Research*, 102:10573–10584, 1997.

[33] P. Fieguth and A. S. Willsky. Fractal estimation using models on multiscale trees. *IEEE Transactions on Signal Processing*, 44(5):1297–1300, May 1996.

[34] P. Fieguth, A. S. Willsky, and W. Karl. Efficient multiresolution counterparts to variational methods for surface reconstruction. *Computer Vision and Image Understanding*, 70(2):157–176, May 1998.

[35] C. Fosgate, H. Krim, W. W. Irving, and A. S. Willsky. Multiscale segmentation and anomaly enhancement of SAR imagery. *IEEE Transactions on Image Processing*, 6(1):7–20, January 1997.

[36] A. Frakt, H. Lev-Ari, and A. S. Willsky. A generalized Levinson-algorithm for covariance extension with application to multiscale autoregressive modeling. *IEEE Transactions on Information Theory*, 49(2), February 2003.

[37] A. Frakt and A. S. Willsky. Computationally efficient stochastic realization for internal multiscale autoregressive models. *Multidimensional Systems and Signal Processing*, 12(2), 2001.

[38] A. B. Frakt. *Internal Multiscale Autoregressive Processes, Stochastic Realization, and Covariance Extension.* PhD thesis, Massachusetts Institute of Technology, August 1999.

[39] A. B. Frakt and A. S. Willsky. Efficient multiscale stochastic realization. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, Seattle, WA, May 1998.

[40] A. B. Frakt and A. S. Willsky. Multiscale autoregressive models and the stochastic realization problem. In *Proceedings of the Asilomar Conference on Signals, Systems, and Computers*, November 1999.

[41] B. J. Frey. *Graphical Models for Machine Learning and Digital Communication.* MIT Press, Cambridge, MA, 1998.

[42] R. G. Gallager. *Information Theory and Reliable Communication.* Wiley, New York, NY, 1968.

[43] M. Golumbic. *Algorithmic Graph Theory and Perfect Graphs.* Academic Press, New York, 1980.

[44] G. R. Grimmett. A theorem about random fields. *Bulletin of the London Mathematical Society*, 5:81–84, 1973.

[45] J. M. Hammersley and P. E. Clifford. Markov fields on finite graphs and lattices. Unpublished, 1971.

[46] P. Heggernes and Y. Villanger. *Lecture Notes in Computer Science*, volume 2461, chapter Efficient Implementation of a Minimal Triangulation Algorithm. Springer-Verlag, January 2002.

[47] T. Ho. *Multiscale Modelling and Estimation of Large-Scale Dynamic Systems.* PhD thesis, Massachusetts Institute of Technology, September 1998.

[48] H. Hotelling. Relations between two sets of variates. *Biometrika*, 28:321–377, 1936.

[49] W. W. Irving. *Multiscale Stochastic Realization and Model Identification with Applications to Large-scale Estimation Problems.* PhD thesis, Massachusetts Institute of Technology, September 1995.

[50] W. W. Irving, W. Karl, and A. S. Willsky. A theory for multiscale stochastic realization. In *Proceedings of the 33rd IEEE Conference on Decision and Control*, volume 1, pages 655–662, Lake Buena Vista, FL, December 1994.

[51] W. W. Irving, L. Novak, and A. S. Willsky. A multiresolution approach to discriminating targets from clutter in SAR imagery. *IEEE Transactions on Aerospace and Electronic Systems*, 33(4):1157–1169, October 1997.

[52] W. W. Irving and A. S. Willsky. A canonical correlations approach to multiscale stochastic realization. *IEEE Transactions on Automatic Control*, 46(10):1514–1528, October 2001.

[53] T. S. Jaakkola and M. I. Jordan. Variational probabilistic inference and the QMR-DT network. *Journal of Artificial Intelligence Research*, 10:291 322, 1999.

[54] F. V. Jensen. Junction trees and decomposable hypergraphs. Technical report, Judex Datasystemer, Aalborg, Denmark, 1988.

[55] F. V. Jensen. *An Introduction to Bayesian Networks.* Springer-Verlag, New York, 1996.

[56] F. V. Jensen and F. Jensen. Optimal junction trees. In *Uncertainty and Artificial Intelligence: Proceedings of the Tenth Conference*, San Mateo, CA, 1994. Morgan Kaufmann.

[57] J. K. Johnson. Estimation of GMRFs by recursive cavity modeling. Master's thesis, Massachusetts Institute of Technology, June 2003.

[58] M. Jordan. *Learning in Graphical Models.* MIT Press, Cambridge, MA, 1999.

[59] M. Jordan. *An Introduction to Graphical Models.* MIT Press, Cambridge, MA, To be published.

[60] R. Kalman and R. Bucy. New results in linear filtering and prediction theory. *The American Society of Mechanical Engineers: Basic Engineering, series D*, 83:95–108, March 1961.

[61] A. Kannan. *Adaptation of spectral trajectory models for LVCSR.* PhD thesis, Boston University, 1997.

[62] A. Kannan and S. Khudanpur. Tree-structured models of parameter dependence for rapid adaptation in large vocabulary conversational speech recognition. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, Phoenix, Arizona, 1999.

[63] A. Kannan and M. Ostendorf. Modeling dependence in adaptation of acoustic models using multiscale tree processes. In *Proceedings of EUROSPEECH*, pages 1863–1866, 1997.

[64] A. Kannan, M. Ostendorf, W. C. Karl, D. A. Castanon, and T. K. Fish. ML parameter estimation of a multiscale stochastic process using the EM algorithm. *IEEE Transactions on Signal Processing*, 48(6):1836–1840, June 2000.

[65] A. Kim. Hierarchical stochastic modeling for segmentation and compression of SAR imagery. Master's thesis, Massachusetts Institute of Technology, June 1997.

[66] A. Kim and H. Krim. Hierarchical stochastic modeling of SAR imagery for segmentation/compression. *IEEE Transactions on Signal Processing*, 47(2):458–468, February 1999.

[67] O. P. Kreidl and A. S. Willsky. Inference with minimal communication: a decision-theoretic variational approach. In *Advances in Neural Information Processing Systems*, volume 18. 2006. To appear.

[68] S. Kullback. *Information Theory and Statistics.* Wiley, New York, 1959.

[69] S. Kullback and R. A. Leibler. On information and sufficiency. *Annals of Mathematical Statistics*, 22(1):79–86, March 1951.

[70] P. Kumar. A multiple scale state-space model for characterizing subgrid scale variability of near-surface soil moisture. *IEEE Transactions on Geoscience and Remote Sensing*, 37(1):182–197, January 1999.

[71] S. L. Lauritzen. *Graphical Models.* Oxford Statistical Science Series. Oxford University Press, 1998.

[72] A. Lindquist and G. Picci. On the stochastic realization problem. *SIAM Journal of Control and Optimization*, 17(3):365–389, May 1979.

[73] M. Luettgen. *Image processing with multiscale stochastic models.* PhD thesis, Massachusetts Institute of Technology, May 1993.

[74] M. Luettgen, W. Karl, and A. S. Willsky. Efficient multiscale regularization with applications to the computation of optical flow. *IEEE Transactions on Image Processing*, 3(1):41–64, January 1994.

[75] M. Luettgen, W. Karl, A. S. Willsky, and R. Tenney. Multiscale representations of Markov random fields. *IEEE Transactions on Signal Processing*, 41(12):3377–3396, December 1993.

[76] M. Luettgen and A. S. Willsky. Likelihood calculation for a class of multiscale stochastic models, with application to texture discrimination. *IEEE Transactions on Image Processing*, 4(2):194-207, February 1995.

[77] M. Luettgen and A. S. Willsky. Multiscale smoothing error models. *IEEE Transactions on Automatic Control*, 40(1):173-175, January 1995.

[78] G. J. MacLachlan and T. Krishnan. *The EM Algorithm and Extensions.* John Wiley and Sons, 1997.

[79] R. J. McEliece, D. J. C. MacKay, and J. F. Cheng. Turbo decoding as an instance of Pearl's belief propagation algorithm. *IEEE Journal on Selected Areas in Communications*, 16(2):140–152, February 1998.

[80] M. Meila. *Learning with mixtures of trees.* PhD thesis, Massachusetts Institute of Technology, 1998.

[81] R. M. Neal and G. E. Hinton. A view of the EM algorithm that justifies incremental, sparse, and other variants. In M.I. Jordan, editor, *Learning in Graphical Models*, pages 355–368. Kluwer Academic Publishers, 1998.

[82] T. Ohtsuki, L. K. Cheung, and T. Fujisawa. Minimal triangulation of a graph and optimal pivoting order in a sparse matrix. *Journal of Mathematical Analysis and Applications*, 54:622–633, 1976.

[83] S. Parter. The use of linear graphs in Gauss elimination. *SIAM Review*, 3:119–130, 1961.

[84] J. Pearl. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference.* Morgan Kaufmann Publishers, Inc., 1988.

[85] J. Pearl. *Causality: Models, Reasoning, and Inference.* Cambridge University Press, 2000.

[86] H. Rauch, F. Tung, and C. Striebel. Maximum likelihood estimates of linear dynamic systems. *AIAA Journal*, 3(8):1445–1450, August 1965.

[87] T. Richardson. The geometry of turbo-decoding dynamics. *IEEE Transactions on Information Theory*, 46(1):9–23, January 2000.

[88] T. Richardson and R. Urbanke. The capacity of low-density parity check codes under message-passing decoding. *IEEE Transactions on Information Theory*, 47:599-618, February 2001.

[89] D. Rose. Triangulated graphs and the elimination process. *Journal of Mathematical Analysis and Applications*, 32:597–609, 1970.

[90] D. Rose, R. Tarjan, and G. Lueker. Algorithmic aspects of vertex elimination on graphs. *SIAM Journal on Computation*, 5(2):266–283, June 1976.

[91] W. Rudin. *Principles of Mathematical Analysis.* McGraw-Hill, 1976.

[92] R. Salakhutdinov, S. Roweis, and Z. Ghahramani. Optimization with EM and expectation-conjugate-gradient. In *Proceedings of the Twentieth International Conference on Machine Learning (ICML-2003)*, Washington DC, 2003.

[93] M. K. Schneider. Multiscale methods for the segmentation of images. Master's thesis, Massachusetts Institute of Technology, June 1996.

[94] M. K. Schneider. *Krylov Subspace Estimation.* PhD thesis, Massachusetts Institute of Technology, February 2001.

[95] M. K. Schneider, P. W. Fieguth, W. C. Karl, and A. S. Willsky. Multiscale methods for the segmentation and reconstruction of signals and images. *IEEE Transactions on Image Processing*, pages 456–468, March 2000.

[96] B. Shipley. *Cause and Correlation in Biology: A Users Guide to Path Analysis, Structural Equations, and Causal Inference.* Cambridge University Press, 2000.

[97] T. P. Speed and H. T. Kiiveri. Gaussian markov distributions over finite graphs. *Annals of Statistics*, 14:138–150, March 1986.

[98] P. Spirtes, C. Glymour, and R. Scheines. *Causation, Prediction, and Search.* MIT Press, Boston, 2000.

[99] N. Srebro. Maximum likelihood markov networks: An algorithmic approach. Master's thesis, Massachusetts Institute of Technology, 2000.

[100] M. Studený and J. Vejnarová. *Learning Graphical Models*, chapter The multiinformation function as a tool for measuring stochastic dependence, pages 261–297. MIT Press, 1999.

[101] E. B. Sudderth. Embedded trees: Estimation of Gaussian processes on graphs with cycles. Master's thesis, Massachusetts Institute of Technology, 2002.

[102] E. B. Sudderth, A. T. Ihler, W. T. Freeman, and A. S. Willsky. Nonparametric belief propagation.

[103] E. B. Sudderth, M. J. Wainwright, and A. S. Willsky. Embedded trees: Estimation of Gaussian processes on graphs with cycles. *IEEE Transactions on Signal Processing*, 52(11):3136–3150, November 2004.

[104] M. J. Wainwright. *Stochastic Processes on Graphs with Cycles: Geometry and Variational Approaches.* PhD thesis, Massachusetts Institute of Technology, January 2002.

[105] M. J. Wainwright, T. S. Jaakkola, and A. S. Willsky. Tree-based reparameterization framework for approximate estimation on graphs with cycles. Technical Report P-2510, LIDS, May 2001.

[106] M. J. Wainwright, T. S. Jaakkola, and A. S. Willsky. Tree-based reparameterization for approximate inference on loopy graphs. In *NIPS 14*. MIT Press, 2002.

[107] M. J. Wainwright, T. S. Jaakkola, and A. S. Willsky. Tree-based reparameterization framework for analysis of sum-product and related algorithms. *IEEE Transactions on Information Theory*, 49(5):1120–1146, May 2003.

[108] D. West. *Introduction to Graph Theory.* Prentice Hall, Upper Saddle River, NJ, 1996.

[109] A. S. Willsky. Multiresolution Markov models for signal and image processing. *Proceedings of the IEEE*, 90(8):1396–1458, August 2002.

[110] C. F. J. Wu. On the convergence properties of the EM algorithm. *The Annals of Statistics*, 11:95–103, 1983.