Graphical Models and Message-Passing Algorithms for Network-Constrained Decision Problems

by

O. Patrick Kreidl

B.S. in Electrical Engineering, George Mason University, 1994
S.M. in Electrical Engineering and Computer Science, M.I.T., 1996

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

February 2008

Signature of Author: _____
Department of Electrical Engineering and Computer Science
January 16, 2008

Certified by: _____
Alan S. Willsky
Edwin Sibley Webster Professor of Electrical Engineering and Computer Science
Thesis Supervisor

Accepted by: _____
Terry P. Orlando
Professor of Electrical Engineering
Chair, Committee for Graduate Students

Graphical Models and Message-Passing Algorithms for Network-Constrained Decision Problems

by O. Patrick Kreidl

Submitted to the Department of Electrical Engineering and Computer Science
on January 16, 2008 in Partial Fulfillment of the Requirements for the Degree
of Doctor of Philosophy in Electrical Engineering and Computer Science

Abstract

Inference problems, typically posed as the computation of summarizing statistics (e.g., marginals, modes, means, likelihoods), arise in a variety of scientific fields and engineering applications. Probabilistic graphical models provide a scalable framework for developing efficient inference methods, such as message-passing algorithms that exploit the conditional independencies encoded by the given graph. Conceptually, this framework extends naturally to a distributed network setting: by associating to each node and edge in the graph a distinct sensor and communication link, respectively, the iterative message-passing algorithms are equivalent to a sequence of purely-local computations and nearest-neighbor communications.

Practically, modern sensor networks can also involve distributed resource constraints beyond those satisfied by existing message-passing algorithms, including e.g., a fixed small number of iterations, the presence of low-rate or unreliable links, or a communication topology that differs from the probabilistic graph. The principal focus of this thesis is to augment the optimization problems from which existing message-passing algorithms are derived, explicitly taking into account that there may be decision-driven processing objectives as well as constraints or costs on available network resources. The resulting problems continue to be NP-hard, in general, but under certain conditions become amenable to an established team-theoretic relaxation technique by which a new class of efficient message-passing algorithms can be derived.

From the academic perspective, this thesis marks the intersection of two lines of active research, namely approximate inference methods for graphical models and decentralized Bayesian methods for multi-sensor detection. The respective primary contributions are new message-passing algorithms for (i) "online" measurement processing in which global decision performance degrades gracefully as network constraints become arbitrarily severe and for (ii) "offline" strategy optimization that remain tractable in a larger class of detection objectives and network constraints than previously considered. From the engineering perspective, the analysis and results of this thesis both expose fundamental issues in distributed sensor systems and advance the development of so-called "self-organizing fusion-layer" protocols compatible with emerging concepts in ad-hoc wireless networking.

Thesis Supervisor: Alan S. Willsky
Title: Edwin Sibley Webster Professor of Electrical Engineering and Computer Science

Acknowledgments

Success is often the result of taking a misstep in the right direction. —*Alan Bernstein*

My path to success in MIT's graduate program is atypical in many ways, including not just the number of years spanned but also the number of people who have helped me along despite my missteps. It is my intention to omit no one but, in light of my growing forgetfulness, I anticipate doing so with probability zero; to those who one day notice having been neglected, I ask for pardon and a chance to make it up to you.

It may be impossible to express the gratitude that I owe to my thesis supervisor, Alan Willsky, whose support and encouragement have entered into every aspect of my well-being over the years. The core ideas in this thesis trace back to deep insights he expressed in our earliest technical conversations. Times of forward progress would be lengthened by his ability to quickly distill my long-winded explanations, always bringing the key points to focus and, ultimately, teaching me the importance of conveying the punch-line first. Times of no progress would be shortened by his willingness to impart wisdom gained from his diverse experiences, always clarifying the nature of academic research and, ultimately, inspiring me to continue giving my best. I strive to one day emulate the composite intellect, energy, efficiency, enthusiasm and compassion he demonstrates as a teacher, researcher, mentor, colleague and friend. (Go Sox & Pats!)

I am grateful to Pablo Parrilo and John Tsitsiklis for their time and service as my thesis readers. Early input provided by John, whose own research from decades ago is drawn upon herein, helped to identify a number of previously unasked questions that are investigated herein. I also wish to thank Pablo, John and (especially) Alan for their flexibility against delays in my delivering drafts for review. Naturally, the responsibility for all remaining typos, oversights or unclear passages rests entirely with me.

I am fortunate to have worked on the fifth-floor of the (in)famous Stata Center, surrounding me with the wonderful people who comprise LIDS and CSAIL. From the pre-Stata days, I wish to acknowledge Michael Athans for bolstering early confidence in my research potential (and many other things since then), Munther Dahleh and Al Drake for bolstering early confidence in my teaching potential (and, again, many other things since then), as well as Sanjoy Mitter for his enduring interest in my total intellectual fulfillment, which includes kindly treating me to numerous coffee breaks and meal outings. Many other faculty members and research scientists notably went out of their way at times on my behalf, including Dimitri Bertsekas, Müjdat Çetin, John Fisher, Vivek Goyal, Eric Grimson, Tommi Jaakkola, Dina Katabi, Muriel Médard, Alexandre Megretski, George Verghese, Jacob White, Karen Willcox and Moe Win.

I benefited greatly from the technical and social interactions with my fellow graduate students. The Stochastic Systems Group (SSG) was a daily source of memorable moments: I give special thanks to Emily Fox, Kush Varshney and Jason Williams for being such delightful officemates; and I thank Venkat Chandrasekaran, Lei Chen, Michael Chen, Jin Choi, Ayres Fan, Jason Johnson, Alex Ihler, Junmo Kim, Dmitry Malioutov,

Mike Siracusa, Erik Sudderth, Walter Sun, Vincent Tan, Andy Tsai and Dewey Tucker for collectively teaching me about too many different things to mention here. I thank Mukul Agarwal, Shashi Borade, Desmond Lun, Sujay Sanghavi and Wee Peng Tay for their pleasant mealtime company, as well as Lav Varshney for being as delightful as one of my officemates. The seventh-floor Control Systems Group (CSG) treated me like one of their own for a number of years: I thank Ola Ayaso, Erin Aylward, Sleiman Itani, Georgios Kotsalis, Jerome Le Ny, Mesrob Ohannessian, Mitra Osqui, Mardavij Roozbehani, Mike Rinehart, Keith Santarelli, Sri Sarma, Pari Shah, Danielle Tarraf and Holly Waisanen-Hatipoglu for including me in their toil and fun. I also thank the organizers of Machine Learning Tea, which similarly added to my technical breadth and also introduced me to many high-quality individuals in that distant other tower.

These rewarding interactions with faculty and other students could not have transpired without the administrative staff that invisibly holds the labs and department together: Lisa Bella, Peggy Carney, Rachel Cohen, Lynne Dell, Doris Inslee, Lisa Gaumond, Brian Jones, Patsy Kline, Michael Lewy, Fifa Monserrate and Marilyn Pierce have all helped create time for me to better perform as a teaching or research assistant.

I am grateful to numerous mentors and colleagues beyond MIT who have supported my missteps between academics and industry. I thank Bill Bennett, Alan Chao, Chris Donohue, Joel Douglas, Greg & Tiffany Frazier, Sol Gully, Hal Jones, Craig Lawrence, Leonard Lublin, Mark Luettgen, Chuck Morefield, Larry Roszman, Nils Sandell and Bob Tenney at (formerly) Alphatech, Inc. for letting me work projects, keeping me connected to the pleasures of industry during my academic years. I thank Guy Beale, Jerry Cook and Andre Manitius at George Mason University for letting me teach courses, keeping me connected to the pleasures of academics during my industry years; throughout all years, the timing of Jerry's uplifting emails has been nothing short of uncanny.

It is definitely impossible to express the gratitude that I owe to my family, particularly my parents (Vic & Alfi) whose infinite love and devotion to my older sisters (Astrid & Sandrine) and me makes this thesis more their accomplishment than my own. Their examples guide me as I continue to question the methods by which I've lived and achieved thus far. I am similarly blessed with the love and examples of my favorite aunt and uncle (Rosi & Jim Stark), whose own children (James, Lisa & Mike, the third being my favorite in-law) are to me like the younger siblings I otherwise would not have. I dedicate the majority of this thesis to these family members—the remaining portion I dedicate to Stephanie Green, whose immediate and enthusiastic exclamation that "going to MIT is the opportunity of a lifetime" was what clinched my misstep to return to academics for good. My heart will forever save a place for her and her family.

I lack the space to list all deserving friends, but let me at least thank Brian Chen, Cara Colgate & Chris Conley, Emily Fox & other SSG skiers, Jenn Lilly & Ed Gelber, Mike Tomlinson as well as Karen Willcox & Jaco Pretorius for the various celebrations. A special thanks is also owed to Ola Ayaso, Mukul Agarwal and Wee Peng Tay for trudging through snow to help improve my defense presentation.

Finally, I gratefully acknowledge the financial support provided by the U.S. Air Force Office of Scientific Research under contract FA9550-04-1-0351 as well as the U.S. Army Research Office under MURI grants W911NF-05-1-0207 and W911NF-06-1-0076.

Contents

List of Figures

Chapter 1

Introduction

PROBLEMS of inferring, estimating or deciding upon the value of a (hidden) random vector based on the observed value of a related random vector are fundamental to a variety of scientific fields and engineering applications. Canonical examples include hypothesis testing in applied statistics [44], spin classification in statistical physics [4], gene phylogeny in molecular biology [25], block decoding in communication theory [32], speech recognition in computer science [85] and texture discrimination in image processing [120]. Seemingly different computational solution methods that appear across these traditionally separated fields can all be studied in the formalism of *probabilistic graphical models* [28, 49, 51, 60, 79, 117, 120]. Graphical models derive their power by combining a parsimonious representation of large random vectors with a precise correspondence between the underlying graph structure and the complexity of computing key summarizing statistics (i.e., marginals, modes, means, likelihoods) to support inference objectives.

When observations are collected by a network of distributed sensors, application of the graphical model formalism may at first seem trivial, as there already exists a natural graph defined by the sensor nodes and the inter-sensor communication structure. Also, the most efficient solutions to key inference problems can be interpreted as iterative message-passing algorithms defined on the graph, featuring a sequence of purely-local computations interleaved with only nearest-neighbor communications and greatly facilitating distributed implementations. However, questions outside the usual lines of inquiry arise if the communication structure implied by the network topology need not be equivalent to the information structure implied by the graphical model. Even otherwise, popular message-passing algorithms (e.g., belief propagation [79]) are derived without consideration for the possibility of decision-driven processing goals or explicit constraints and costs on available network resources (e.g., computation cycles, communication bandwidths). Such issues have already inspired inquiries into the

robustness of existing message-passing algorithms to unmodeled resource constraints [18, 24, 47, 77, 78, 90, 95], demonstrating limits to their reliability and motivating alternative distributed solutions that will degrade gracefully even as network constraints become severe.

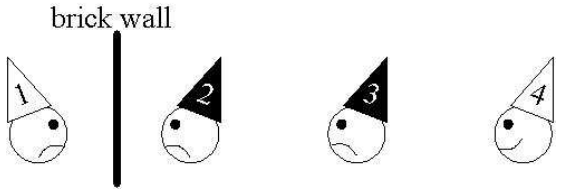
This thesis focuses on an important class of *network-constrained decision problems*, the key challenge being that the information structure and the network constraints are generally defined by two different graphs. One graph underlies the probabilistic model that jointly describes all sensors' hidden and observable random variables, while the other graph underlies the communication model that renders the usual graph-based message-passing algorithms infeasible or unreliable. For example, assuming the special case where the two graphs are identical, it is well known that the popular belief propagation algorithm ideally requires communication overhead of *at least* two real-valued messages per edge. In contrast, our class of problems moves towards having to compress, or quantize, these messages such that total communication overhead is *at most* a fixed number of finite-alphabet *symbols* (e.g., two "bits" per edge). Goals of processing (i.e., decisions to be made by some or all sensors) need to be taken into account to make best use of these limited bits. The necessary departure from existing message-passing solutions only becomes more pronounced when the communication graph may differ from the probability graph.

1.1 Motivation

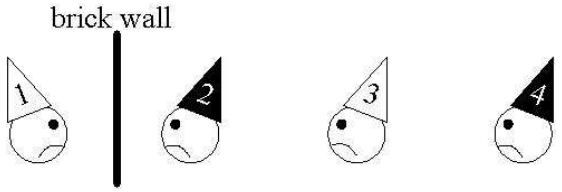
1.1.1 A Simple Puzzler

Figure 1.1 depicts a simplest instance of the class of network-constrained decision problems addressed in this thesis. Four hats, two colored white and two colored black, are randomly assigned to four different nodes. Each node is able to observe only the hats in its forward view, yet no node is able to observe beyond the brick wall; that is, nodes one and two observe only the wall, while node three observes hat two and node four observes both hats two and three. The decision objective is that exactly one node calls out the correct color of its own hat, and the network constraint is that no one node communicates to another (except via the final call).

The problem would be trivial without the network constraint e.g., elect node three as the leader, making the correct call after node four has communicated the observed color of hat three. With the network constraint, however, it is not immediately apparent that there exists a feasible solution with error-free performance. Specifically, if we elect



(a) Random hat configuration in which node four can always make the correct call.



(b) Random hat configuration in which no one node can always make the correct call.

Figure 1.1. Illustration of the two types of hat configurations that arise in the simple network-constrained inference problem considered in Section 1.1. The configuration in (a), in which case node four can always make the correct call, occurs only one-third of the time. Two-thirds of the time node four faces the hat configuration in (b), in which case its call amounts to a blind guess.

nodes one or two to make the call, each knowing only that its own hat is equally-likely to be white or black, there is 50% chance of error. If node three makes the call, knowing that there are exactly two hats of each color and thus always choosing the opposite color of hat two, there is 33% chance of error. Electing node four to make the call also leads to 33% chance of error: while the correct call is easily made when node four faces the hat configuration of Figure 1.1(a), two-thirds of the time node four faces the hat configuration of Figure 1.1(b) and its information degenerates to that of nodes one or two.

The best solution, feasible yet also achieving zero error, is for nodes three and four to cooperatively exercise a leadership role; that is, if node four makes the call only when facing the hat configuration in Figure 1.1(a), then upon *not* hearing from node four, node three can deduce its own hat must be different from hat two and itself make the correct call. Note that the selective silence, resourcefully communicating one bit of information from node four to node three, is maximally informative only because all of the nodes *a-priori* agree on its meaning in the global context of the probabilistic model and inference objective; that is, not only must nodes three and four appropriately coordinate their leadership roles, nodes one and two must also agree to never enter into their respective leadership roles. Stated more generally, to maintain satisfactory decision performance subject to severe network constraints, *every* node must acquire a

fairly rich understanding of the global problem before it can determine a local rule that is simultaneously resourceful and informative to the *team* [42, 86].

■ 1.1.2 Collaborative Self-Organizing Sensor Networks

The vision of collaborative self-organizing sensor networks, a confluence of emerging technology in both miniaturized devices and wireless communications, is important to numerous scientific fields and engineering applications e.g., geology, biology, surveillance, fault-monitoring [19, 34, 45, 87, 92, 124]. Their promising feature is the opportunity for each spatially-distributed sensor node to receive measurements from its local environment and transmit information that is relevant for effective global decision-making. No matter the specific application, because each node possesses only finite battery power, the design of a network-wide measurement processing strategy faces an inherent yet complex tradeoff between maximizing the application-layer global objective of decision-making performance and the network-layer global objective of energy efficiency.

Most classical decision-theoretic problem formulations are agnostic about explicit constraints or costs on algorithmic resources (e.g., computation cycles, communication bandwidths). In turn, while perhaps providing a useful benchmark on achievable decision-making performance, a classically-derived measurement processing strategy is unlikely to admit an energy-efficient distributed implementation. Conversely, suppose each node implements a local measurement processing rule that is classically-derived as if assuming complete isolation from all peripheral nodes. Then, especially when local measurements are strongly correlated and constraints on computation or inter-node communication are severe, the resulting network-wide strategy may become overly myopic, in the sense that the achieved decision-making performance is unsatisfactory for the application at hand long before the end of the network's operational lifetime.

A wireless sensor network therefore befits a measurement processing strategy that is both optimized for application-layer decision performance and subject to energy-based constraints dictated by the network layer. On the other hand, because an ad-hoc network is anticipated to repeatedly self-organize (e.g., to stay connected due to node dropouts, link failures, etc.) over its lifetime, we should anticipate having to repeatedly re-optimize the network-constrained strategy. So, unless this *offline* optimization algorithm is itself amenable to an energy-efficient distributed implementation, there is little hope for maintaining application-layer decision objectives without also rapidly diminishing the network-layer resources that remain for actual *online* measurement pro-

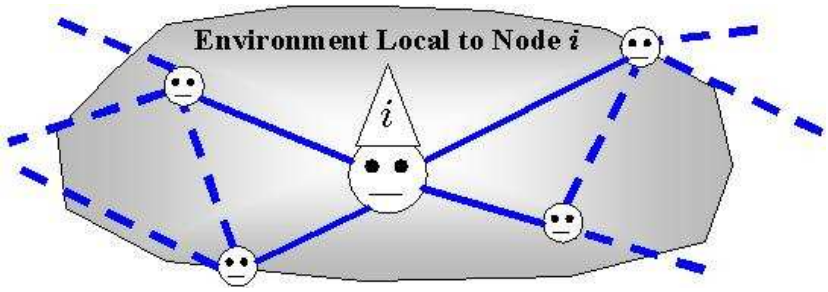


Figure 1.2. Extrapolation of the simple hat problem depicted in Figure 1.1 in ways motivated by sensor network applications. Each sensor node i receives noisy measurements from its local environment, may share compressed information over low-rate or unreliable communication links with its neighbors and, ultimately, may form its own local state estimates. However, these spatially-distributed nodes are generally initialized without detailed knowledge of the environment beyond its nearest neighbors, suggesting some amount of costly communication is essential. The core design problems arise due to the competing global objectives of maximizing application-layer decision performance and maximizing network-layer energy efficiency.

cessing. In particular, it must be that the price of performing these intermittent offline optimizations can be amortized over a substantial number of online usages, so that the total network resources consumed for offline purposes still represents only a modest fraction of the resources available over the total operational lifetime.

Figure 1.2 extrapolates from the simple hat problem discussed in the preceding subsection in ways motivated by emerging concepts in wireless sensor networks. The random hat configurations correspond to states of the global environment, each spatially-distributed node receiving a noisy measurement related only to the state of its local environment. A node calling out its own hat color corresponds to a sensor providing an estimate of its local state to the network "gateway," which in general can include multiple or even all sensor nodes (and any node not in this gateway is thus a "communication-only" node). The dominant resource constraints are assumed to arise from the underlying communication medium, the network topology defined by a graph with each edge representing a point-to-point, low-rate link between two nodes. Every active symbol transmission consumes significant power, incentivising each node to use the links with its neighbors selectively, and the multipoint-to-point link into each node from its neighbors can be unreliable e.g., due to uncoded interference or packet loss.

Especially in sparsely-connected sensor networks, it is clear that some online communication, whether costly or unreliable, is required if each gateway decision is to have any hope of accounting for valuable information observed by communication-only

nodes. Even so, the central offline design questions are analogous to those explored in the simple hat problem. How do the distributed nodes collectively identify and resourcefully disseminate this valuable information? What are the achievable tradeoffs between the two competing global objectives of minimal "gateway node-error-rate" and minimal "networkwide link-use-rate?"

■ 1.2 Principal Research Areas and Thesis Contributions

The overarching objective in this thesis is to characterize the most resourceful distributed algorithmic solutions possible for the class of network-constrained decision problems motivated in the preceding section. The distinguishing assumption from their unconstrained counterparts is the non-ideal communication model, which includes the possibilities of finite-rate, unreliable links and a network topology different from the graph structure underlying the probabilistic model. Explicit constraints and costs on communication resources, especially if severe, fundamentally alter the character of satisfactory solution methods. For instance, the canonical inference challenge of finding efficient yet convergent message-passing approximations for *online* estimation in "loopy" graphical models is met trivially by constraint. The key challenges rather arise in finding tractable distributed solutions to the associated *offline* design problems, seeking to preserve satisfactory decision performance no matter the explicit online constraints.

The team-theoretic approach to network-constrained decision problems described in this thesis both draws from and contributes to the intersection of two established research areas, namely *approximate inference methods in graphical models* [28, 49, 51, 60, 79, 117, 120] and *decentralized Bayesian methods in multi-sensor detection* [11, 106, 109, 110]. Our problem formulation leverages the former primarily for compact representations of the probabilistic model and the latter primarily for non-ideal representations of the communication model. Our solution methods contribute, respectively,

- new quantized message-passing algorithms for online estimation in which global decision performance degrades gracefully as network constraints become arbitrarily severe and
- new efficient message-passing interpretations for offline optimization that remain tractable in a larger class of decision objectives and network constraints than previously considered.

Our distinction between online processing and offline optimization underscores a fundamental consideration for engineering collaborative self-organizing sensor networks: if

self-organization procedures are expected to maintain application-layer decision performance in the face of network-layer resource constraints, then assessing the value and feasibility of design alternatives cannot neglect the fact that such procedures will themselves consume some of the resources that the stipulated constraints aim to conserve.

We now survey these two principal research areas, also elaborating upon both their influence on our approach and our advances on their state-of-the-art.

■ 1.2.1 Approximate Inference Methods in Graphical Models

Many important inference problems can be posed as the computation of certain summarizing statistics (e.g., marginals, modes, means, likelihoods) given a multi-variate probability distribution, where some variables are measured while others must be estimated based on the observed measurements. The practical challenges stem from the fact that, in general, the representation and manipulation of a joint probability distribution scales exponentially with the number of random variables being described. The graphical model formalism [28, 49, 51, 60, 79, 117, 120] provides both a compact representation of large multivariate distributions and a systematic characterization of the associated probabilistic structure to be exploited for computational efficiency. Fundamentally, a graphical model represents a family of probability distributions on the underlying graph: nodes are identified with random variables and edges (or the lack thereof) encode Markov properties among subsets of random variables. Indeed, the formalized idea of exploiting Markov structure for computational efficiency is evident in a number of applied statistical fields e.g., Ising models in physics [4], low-density parity check codes in communications [31], hidden Markov models in speech recognition [85], multi-resolution models in image processing [120].

A compact representation of joint probability distributions is, by itself, not sufficient to tractably solve large-scale inference problems. The complexity of inference given a graphical model also depends strongly on the underlying graph, where the fundamental divide is whether it contains cycles. For graphs without cycles, or *trees*, direct computation of many important summarizing statistics can be organized recursively in a manner that scales linearly in the number of nodes [120]. The many variants of this basic idea comprise the class of graph-based message-passing algorithms broadly lumped under the term of *belief propagation* [28, 51, 79]. Belief propagation algorithms essentially amount to iterating over a certain set of nonlinear fixed-point equations [6, 71], relating the desired inference solution to so-called *messages* passed between every node and its immediate neighbors in the graph. Such iterations always converge in a tree-structured

graphical model, the final messages into each node representing sufficient statistics of the information at all other nodes. The *junction-tree* algorithm [49] is a generalization of these iterative solution methods, resting upon a precise procedure for adding edges and aggregating nodes to convert the original cyclic, or *loopy*, graphical model to an equivalent tree-structured model. This technique exposes that optimal inference in graphical models remains tractable only for graphs with narrow tree-width i.e., cyclic graphs in which only a relatively small number of nodes need to be aggregated to form the equivalent junction tree.

Many practical applications, of course, give rise to graphical models for which exact inference is computationally infeasible. Variational methods for approximate inference start by expressing the intractable solution as the minimizing (or maximizing) argument of a mathematical optimization problem [50, 52, 113, 123]. One can often recover existing algorithms from different specializations of such an optimization problem. More importantly, by relaxing or otherwise modifying this optimization problem to render it amenable to mathematical programming techniques [5, 6, 7, 9], one can obtain tractable yet effective approximations to the original inference problem and, ideally, an analysis of error bounds or other fundamental limits associated with alternative approximations.

Variational methods have recently been the vehicle towards an improved understanding of the popular *loopy* belief propagation (BP) algorithms (see [113] for a broad view of these ideas). Originally a heuristic proposal to simply iterate the BP fixed-point equations as if the underlying graph were free of cycles [79], the efficient algorithm (if it converged) found considerable empirical success in a variety of large-scale practical applications [29, 53, 57, 65, 70]. Early theoretical explorations into its convergence and correctness properties considered special-case cyclic structures (e.g., a single-cycle graph [114] or the limit of infinite-length cycles [89]). Variational interpretations uncovered links between loopy BP and the rich class of entropy-based *Bethe free energy* approximations in statistical physics, collectively establishing that every graphical model has at least one BP fixed point [99, 123], that stable BP fixed points are local minima of this free energy [39, 116, 123], sufficient conditions for uniqueness of BP fixed-points [40, 47, 99], several different characterizations of the convergence dynamics [47, 63, 67, 94, 111], as well as algorithmic extensions based on higher-order entropy-based approximations [66, 115, 123] and connections to information geometry and convex programming [112, 113, 118].

The variational methods in this thesis forge a sharp departure from belief propagation (BP). Firstly, motivated by sensor network applications, we return to BP's

traditional message-passing view, assuming that the nodes in the graph physically correspond to spatially distributed sensors/processors. Secondly, our need for approximation is dominated by (typically severe) constraints on the available *communication* resources, assuming a low-rate or unreliable network with topology not necessarily related to the given probability graph. Efficient computation remains a concern as well: in particular, we essentially bypass the issue of convergence by allowing from the start only a fixed small number of message-passing iterations. Altogether, our variational formulation expresses the need to essentially redesign the online measurement processing algorithm subject to the explicit network constraints. Also in contrast to other variational methods, our approximation is driven by decision-based objectives (as opposed to entropy-based objectives) that may also capture costs associated to communication-related decisions, which tune our network-constrained solutions for more focused high-level goals than the relatively generic processing goals of traditional message-passing solutions. Even so, certain special cases of our formulation allow for approximations of these more generic statistical quantities (e.g., posterior node marginals or data likelihoods), should they also be of direct interest.

Other recent works in approximate inference, also looking towards distributed sensing applications, consider communication-constrained variants of graph-based message-passing algorithms. An experimental implementation of belief propagation within an actual sensor network concludes that reliable communications are indeed the dominant drain on battery power, with overhead varying substantially over different message schedules and network topologies [24]. Also in the context of sensor networks, a modification of the junction-tree algorithm introduces redundant representations to compensate for anticipated packet losses and node dropouts [77, 78]. Some theoretical impacts of finite-rate links in belief propagation have also been addressed [47, 90], essentially proving that "small-enough" quantization errors do not alter the behavior of BP algorithms. A similar robustness property is observed empirically in an application of belief propagation to distributed target tracking problems, where "occasionally" suppressing the transmission of a message is shown to have negligible impact on performance and, in some cases, can even speed up convergence [18]. Conceptually, these views on communication constraints relate closely to the general problem of BP message approximation [47, 53, 66, 95], which generically arises due to the infinite-dimensional messages implied by BP in the case of (non-Gaussian) continuous-variable graphical models.

The network communication constraints considered in this thesis depart significantly from those found in existing belief propagation (BP) algorithms. In contrast to

proposing modifications directly to the BP algorithms, we explicitly model the network constraints inside an otherwise unconstrained formulation by which the algorithms can be derived. Then, via analysis of the resulting constrained optimization problem, we can examine the extent to which different processing algorithms mitigate the *loss* from optimal performance subject to the network constraints. While still conceptually related to the problem of BP message quantization, especially when the communication and probability graphs happen to coincide, our consideration for the "low-rate" regime appears to be unique.

■ 1.2.2 Decentralized Bayesian Methods in Multi-Sensor Detection

Classical single-sensor detection, or hypothesis testing, is perhaps the most elementary decision problem under uncertainty [108]. The true state of the environment is not fully observable but a sensor, upon receiving a noisy measurement, must generate a state-related decision without delay. Subject to design is the *decision rule*, or the function or algorithm, by which any particular measurement is mapped to a particular decision. The choice of decision rule clearly depends on both a probabilistic model of the uncertain environment and the criterion by which one quantifies the rule-dependent decision performance. The basic problem has been studied under a number of different decision criteria e.g., Neyman-Pearson [108], Ali-Silvey distances [82], mutual information [30], robust/minimax [46]. This thesis focuses exclusively on the canonical minimum-Bayes-risk criterion, a special case of which is the basic error probability criterion [108].

Though formally posed as a minimization over a function space, in which optimality is generally intractable [10, 73], the single-sensor Bayesian detection problem admits a straightforward analytical simplification called the *likelihood-ratio test*. The problem's decentralized counterpart [11, 106, 109, 110] was formally introduced in [104] for the special case of a binary hypothesis test with two distributed sensor nodes. Taking a *team*-theoretic perspective [42, 64], which assumes the nodes agree on a common organizational objective but will generate local decisions based on different information, the solution was expressed as a pair of likelihood-ratio tests with appropriately coupled threshold values. This initial analysis required a certain statistical independence assumption, later established to be essential for analytical tractability: in general, even for just two nodes, the problem of optimal decentralized detection is proven to be NP-complete [107]. A related implication is that the optimal decentralized strategy, again in general, need not lie within a finitely-parameterized subset of the function space defined by all *feasible* online processing strategies [48, 119, 125].

Assuming conditional independence, computing the coupled likelihood-ratio tests boils down to solving a system of nonlinear equations, each expressing one sensor's threshold as a function of the (global) probabilistic model, decision objective and the other sensor's threshold. The natural algorithm for solving these equations starts with an arbitrary initial set of thresholds and iterates the equations on a sensor-by-sensor basis (i.e., a Gauss-Seidel iterative solution [6, 71]). The (generally non-unique) fixed points of this iterative algorithm correspond to different so-called *person-by-person optimal* processing strategies in team theory [42, 64, 106], each known to satisfy necessary (but not always sufficient) optimality conditions of the original problem. That is, while the set of all fixed points will contain the globally-optimal decentralized strategy, it also contains any number of local optima or saddle-points in the absence of additional convexity assumptions [6, 41]. Nevertheless, the correspondence between person-by-person optimality conditions and cyclically iterating the fixed-point equations guarantees the sequence of strategies monotonically improves (or at least never degrades) the global decision performance.

These fundamental analytical and algorithmic results readily extend to a variety of detection networks involving more than two sensors as well as inter-sensor communication. The canonical formulation considers a set of sensors collectively solving a binary hypothesis test, where each receives its own local measurement and transmits a binary-valued signal to a common "fusion center" responsible for making the final (team) decision [21, 26, 38, 43, 98, 106, 109, 110]. Called the parallel (or fusion) *architecture*, referring to the graph structure underlying the communication model, a person-by-person optimal strategy is generally seen to introduce asymmetry across the local processing rules e.g., even if all remote sensors have identical noise characteristics, the fusion center generally benefits when they employ the correct combination of non-identical local rules. Analogous results exist for a series (or tandem) architecture, where the sensor at the end of the line makes the final team decision, and tree architectures, where the root sensor makes the final team decision [26, 96, 97].

The decision architectures considered in this thesis include many of the ones considered in previous work, certainly those mentioned above, as special cases. Firstly, our analysis applies to any directed acyclic architecture, reducing the person-by-person optimality conditions to a finite-dimensional fixed-point equation (assuming conditional independence, of course). This reduction was previously thought to be possible only for tree-structured networks [97, 106, 110] or, in the case of general directed acyclic networks, alongside additional simplifying assumptions on the probabilistic model and

decision objectives [80, 81]. The generality of our formulation and proof technique also recovers numerous other extensions examined separately in the literature, including a vector state process along with a distributed decision objective [80, 81] (versus only a global binary hypothesis test by a designated fusion center), as well as selective or unreliable online communication [16, 17, 74, 83, 88] (e.g., a sensor may opt to suppress a costly transmission, a link may experience a symbol erasure). Our generality also affords extensions to undirected and hybrid architectures, respectively allowing for bidirectional and (perhaps costly) long-distance communication, as well as to multi-stage communication architectures. The associated team-theoretic analyses provide new structural results that complement an existing class of decentralized processing solutions for sequential binary detection problems [3, 36, 72, 103]. Finally, experiments throughout the thesis also add to the understanding of fundamental tradeoffs between performance and communication with respect to architecture selection [37, 76].

Other recent work in decentralized binary detection, also looking towards sensor network applications, steers away entirely from the team-optimal algorithmic solution discussed above [1, 15, 75, 101, 102, 105, 122]. The reasons cited include the worst-case NP-completeness result and the (correct) recognition that, even with conditional independence, the convergent offline algorithm generally requires that (i) all nodes are initialized with a consistent global picture of both the uncertain environment and the decision objectives, and (ii) iterative per-node computation (and offline communication) overhead scales exponentially with the number of nodes. Instead, this other recent work focuses on understanding performance and communication tradeoffs across different classes of asymptotic approximations, each based on the limit of an infinite number of nodes under assumptions of network regularity and sensor homogeneity.

The offline iterative algorithms developed in this thesis can be viewed as "best-case" solutions to the team-optimal fixed-point equations. The generality of our proof technique exposes special structure associated with the communication model, analogous to that associated with the probabilistic model in the derivation of belief propagation algorithms. Taken in combination with additional model assumptions (which include conditional independence), we discover that the offline algorithm admits an efficient message-passing interpretation; each node need only be initialized with a local picture of the uncertain environment and decision objectives, while iterative per-node overhead becomes invariant to the number of nodes (but still scales exponentially with the number of neighbors, so large networks are taken to be sparsely connected). In the well-studied case of binary hypothesis testing in directed networks with a designated

fusion center, our algorithm specializes to a known efficient algorithm [96, 97], but we note that our derivation does not depend on a differentiable measure of decision performance nor on quantities tied to binary detection. This special case of our algorithm also complements the recent work on asymptotic approximations mentioned above, offering a tractable design alternative when assumptions of network regularity or sensor homogeneity cannot be made. Our offline message-passing algorithm also generalizes known computational methods for the case of a structured state process [80, 81], in the sense that we guarantee efficiency and correctness without assuming that the two graphs be the same. The extensions of our message-passing solution to the cases of undirected networks and multi-stage architectures appear to be unique.[1]

■ 1.3 Thesis Organization

The overarching hypothesis driving this thesis is that fully distributed algorithmic solutions for an important class of network-constrained decision problems can be found at the intersection of two traditionally separated areas of active research, namely approximate inference methods in graphical models and decentralized Bayesian methods in multi-sensor detection. The former provides a compact representation of spatially-distributed random vectors and focuses on the tractable computation of key summarizing statistics, but the possibility of explicit (and typically severe) constraints/costs on communication resources is largely unaddressed. The latter folds in a non-ideal communication model and the possibility of higher-level decision-making goals at the start, but to preserve satisfactory (online) performance depends upon the (offline) solution to a generally intractable constrained optimization problem. We reconcile the contrasting perspectives of these two research areas, fostering strong support for our hypothesis, in the remainder of this thesis: its chapter-by-chapter organization is as follows.

Chapter 2: Background
This chapter contains the background underlying the developments in the remainder of this thesis. It first reviews notational conventions and other basic concepts in graph theory and probability theory. These concepts are used to describe the two principal mathematical models that inspire the problems to be formulated and analyzed in subsequent chapters. For Bayesian detection models, we discuss the classical single-sensor formulation and its optimal solution. The natural generalization to multi-sensor problems suggests two baseline decision strategies: the optimal centralized strategy,

[1] The message-passing algorithms discussed in this paragraph have been published [54, 55, 56, 121].

having no regard for possible communication constraints, and the myopic decentralized strategy, satisfying the extreme constraint of zero communication overhead. For probabilistic graphical models, we discuss pairwise discrete representations and a couple of different message-passing algorithms for efficient online estimation. Connections are made between the optimal centralized detector and several different online estimation problems that can be posed given a graphical model. Even at the introductory level, the inherent complexity that drives the active research interest in approximate inference, and the necessary departure from existing message-passing solutions in the face of explicit communication constraints, both become apparent.

Chapter 3: Directed Network Constraints

This chapter describes the team-theoretic solution approach for network constraints in which only unidirectional inter-sensor communication is assumed. Specifically, no matter the graph structure underlying the probabilistic model, we assume the nodes communicate in succession according to a given directed acyclic graph, each node transmitting at most one finite-alphabet symbol (per local measurement). The constrained optimization problem proposed here extends the canonical decentralized detection problem in a number of ways: first, each sensor's measurement relates only to its local state, which is itself correlated with the states local to all other nodes; second, each node can employ a selective, or censored, transmission scheme (i.e., each sensor may, per outgoing link, exercise a cost-free "no-send" option); and, third, the multipoint-to-point channel into each node can be unreliable (e.g., due to uncoded interference or packet loss). Existing team theory establishes when necessary optimality conditions reduce to a convergent iterative algorithm to be executed offline. While the resulting online strategy is efficient by design, this most-general offline algorithm is seen to have exponential complexity in the number of nodes and its distributed implementation assumes a fully-connected network.

We state conditions under which the offline algorithm admits an efficient message-passing interpretation, featuring linear complexity in the number of nodes and a natural distributed implementation. Specifically, the algorithm can be viewed as performing repeated forward-backward sweeps though the given network: each forward sweep propagates "likelihood" messages, encoding what online communication along each link means from the transmitter's perspective, while each backward sweep propagates "cost-to-go" messages, encoding what online communication along each link means from the receiver's perspective. In each offline iteration, both types of incoming messages in-

fluence how each node updates its local rule parameters before it engages in the next iteration. We apply the efficient message-passing algorithm in experiments with a simulated network of binary detectors, characterizing the achievable tradeoff between global detection performance and networkwide online communication in a variety of scenarios. The empirical analysis reveals that, considering the severity of the online communication constraints, relatively dramatic improvements over myopic decentralized performance are possible. In addition, the team strategies are observed to resourcefully attach value to remaining silent, essentially conveying an extra half-bit of information per link even in the presence of faulty channels and cost-free communication. Our empirical analysis also exposes a design tradeoff between constraining in-network processing to conserve algorithmic resources (per online measurement) but then having to consume resources (per offline organization) for the sensors to maintain satisfactory decision performance subject to these constraints.

Chapter 4: Undirected Network Constraints

This chapter develops the team-theoretic solution approach for network constraints defined by an undirected graph, each edge representing a bidirectional (and perhaps unreliable) finite-rate communication link between two distributed sensor nodes. Every node operates in parallel, processing any particular local measurement in two (discrete) decision stages: the first selects the symbols (if any) transmitted to its immediate neighbors and the second, upon receiving the symbols (or lack thereof) from neighbors, decides the estimate of its local state. Our analysis proves that, relative to the analysis for directed networks in Chapter 3, the model requires more restrictive assumptions to avoid worst-case offline complexity, yet less restrictive assumptions to attain best-case offline efficiency. The offline message-passing algorithm translates into a two-stage parallel schedule on the undirected network, where the nodes alternate between exchanging "likelihood" messages (followed by updates to local detection rules) and exchanging "cost-to-go" messages (followed by updates to local communication rules). We assess empirically the performance of the undirected message-passing algorithm, using essentially the same models and setup used in the experiments of Chapter 3.

Architecturally, our analysis and experiments suggest a directed network is preferable when only a few nodes are to provide state estimates (and these nodes are at the end of the succession implied by the directed graph), and an undirected network is preferable when all nodes are to provide state estimates. We also examine the prospect of hybrid network constraints to improve performance in problems for which neither

network alone is satisfactory. We specifically consider two hierarchical decision architectures relevant to sensor network applications, each assuming a subset of the nodes are capable of "long-distance" communication amongst themselves in between communications with their spatially-local neighbors. We show that, in each hierarchical decision architecture, team-optimality conditions are satisfied by an appropriate combination of the directed and undirected offline message-passing algorithms.

Chapter 5: On Multi-Stage Communication Architectures

The two preceding chapters focus on network-constrained decision architectures in which there is only a single-stage of online communication; this chapter aims to generalize the formulation, analyses and results to allow for multiple online communication stages. The multi-stage architectures we consider take their inspiration from the canonical message-passing algorithms that exist for probabilistic graphical models, where we formulate repeated forward-backward sweeps given a directed network and repeated parallel exchanges given an undirected network within a common mathematical framework. Of course, as in our single-stage formulations, the online network is constrained to low-rate or unreliable links and the associated communication graph need not be equivalent to the probability graph. We then apply the team-theoretic analysis of previous chapters, exposing a number of new structural properties that an optimal multi-stage decision strategy should satisfy. These include the minimal assumptions under which online computation grows linearly with the number of nodes (given a sparsely-connected communication graph). Moreover, we show how each local processing rule can make explicit use of memory, affording each node an increasingly accurate sequence of decisions as a function of *all* previously observed information (i.e., all symbols the node has both received and transmitted in earlier communication stages).

Even under best-case model assumptions, however, the required memory and, in turn, the offline solution complexity scales exponentially with the number of online communication stages, necessitating additional approximations. We describe one such approximate offline algorithm, leveraging the offline message-passing algorithms derived in preceding chapters. The key idea is to limit the look-ahead of each node when designing its multi-stage communication rule, but then compensate for their collective sub-optimality via our analytical result for the optimal structure of the final-stage detection rules. A number of small-scale experiments with this approximation indicate that near-optimal detection performance is achievable (despite the constraint to ternary-valued symbols) in a number of communication stages comparable to the diameter of

the network. These experiments also include direct comparisons with approximations based on the (unconstrained) belief propagation algorithm, demonstrating our network-constrained online strategies can yield reliable solutions even in so-called "frustrated" graphical models when belief propagation (BP) often fails to converge. Altogether, the results of this chapter provide the first inroads into the difficult problem of BP message quantization in the "low-rate" regime.

Chapter 6: Conclusion

This chapter summarizes the contributions of this thesis and identifies a number of open questions for future research. The fundamental divide in complexity between single-stage and multi-stage decision architectures exposed during the course of this thesis is evident in both our contributions summary and our future work recommendations.

Chapter 2

Background

THIS chapter summarizes the essential background to understand the problem formulations and solution algorithms presented in subsequent chapters. Section 2.1 starts with a self-contained primer on the mathematical subjects of graph theory and probability theory, both fundamental to the two principal models reviewed in the remaining sections. For Bayesian detection models (Section 2.2), we discuss the classical single-sensor formulation and its well-known optimal solution. The natural generalization to multi-sensor problems suggests two baseline decision strategies: the optimal centralized strategy, having no regard for possible communication constraints, and the myopic decentralized strategy, satisfying the extreme constraint of zero communication overhead. For probabilistic graphical models (Section 2.3), we discuss pairwise discrete representations and a couple of different message-passing algorithms for efficient online estimation. Throughout, key definitions and concepts are illustrated by examples, many of which will also be used in experiments described in future chapters.

■ 2.1 Mathematical Preliminaries

We begin by reviewing basic terminology and notational conventions from both graph theory and probability theory used in this thesis (more detailed introductions appear in e.g., [12] and [8], respectively). Both make use of standard set theory [68], and we sometimes employ a short-hand notation for certain set differences. Specifically, let $\mathcal{V}$ be any set and consider any two subsets $\mathcal{V}_1, \mathcal{V}_2 \subset \mathcal{V}$. The notation $\mathcal{V}_1 \backslash \mathcal{V}_2$ is equivalent to the set difference $\mathcal{V}_1 - \mathcal{V}_2$. We may write $\mathcal{V}_1 \backslash v$ in the special case that $\mathcal{V}_2$ is a singleton set $\{v\}$ for some $v \in \mathcal{V}$, and $\backslash v$ if it is also the case that $\mathcal{V}_1 = \mathcal{V}$.

■ 2.1.1 Elements of Graph Theory

A *graph* $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ is defined by a finite set of nodes, or *vertices*, $\mathcal{V} = \{1, \ldots, n\}$ and a set of node pairs, or *edges* $\mathcal{E} \subset \mathcal{V} \times \mathcal{V}$. We focus only on *simple* graphs for which an edge from any node back to itself and duplicate edges can both be omitted from edge set $\mathcal{E}$ without loss of generality. Simple graphs can be *undirected* or *directed*, the edge set of the former being any subset of the (unordered) node pairs $\{\{i, j\} \mid i \in \mathcal{V} \text{ and } j \in \{i+1, \ldots, n\}\}$, while the edge set of the latter being any subset of the (ordered) node pairs $\{(i, j) \mid i \in \mathcal{V} \text{ and } j \in \mathcal{V} \backslash i\}$. These different edge sets emphasize that node pairs (i, j) and (j, i) denote different edges in a directed graph, but the same edge $\{i, j\}$ in an undirected graph. We say that $\{i, j\}$ is the *undirected counterpart* to the directed edge (i, j) or (j, i). In definitions that apply to either type of graph, our convention is to use the finer directed edge notation but where, in the case of an undirected graph, each (i, j) or (j, i) is understood to indicate the undirected counterpart $\{i, j\}$.

The set of *neighbors*, or *open neighborhood*, of node i in graph $\mathcal{G}$ refers to its adjacent nodes defined by

$$ne(i) = \{j \in \mathcal{V} \mid (i, j) \in \mathcal{E} \text{ or } (j, i) \in \mathcal{E}\}.$$

The *closed neighborhood* of node i in $\mathcal{G}$ is the union $ne(i) \bigcup \{i\}$. The *degree* of node i in $\mathcal{G}$ is the number of neighbors $|ne(i)|$.

Define a *path* as any graph with edge set of the form $\{(v_1, v_2), (v_2, v_3), \ldots, (v_{n-1}, v_n)\}$, where $v_1, v_2, \ldots, v_n$ denote some permutation of its node set. We see that all nodes in a path have degree two, except for the *endpoint nodes* $\{v_1, v_n\}$ which each have degree one. We say that $\mathcal{G}$ is a length-n path from v_1 to v_n. A *cycle* is any path with the additional edge (v_n, v_1). A length-n cycle can be viewed as a length-n path from any node back to itself, except for the resulting absence of uniquely defined endpoint nodes.

We say that $\mathcal{G}' = (\mathcal{V}', \mathcal{E}')$ is a *subgraph* of $\mathcal{G}$ if $\mathcal{V}' \subset \mathcal{V}$ and $\mathcal{E}' \subset \mathcal{E}$. If $\mathcal{V}' = \mathcal{V}$, then $\mathcal{G}'$ is called a *spanning* subgraph of $\mathcal{G}$. A subgraph of $\mathcal{G}$ that is itself a path (cycle) is said to be a path (cycle) in $\mathcal{G}$. If $\mathcal{V}_1$, $\mathcal{V}_2$ and $\mathcal{V}_3$ denote three disjoint subsets of $\mathcal{V}$, then the set $\mathcal{V}_2$ is said to *separate* sets $\mathcal{V}_1$ and $\mathcal{V}_3$ if every path in $\mathcal{G}$ between a node in $\mathcal{V}_1$ and a node in $\mathcal{V}_3$ passes through a node in $\mathcal{V}_2$.

Undirected Graphs

Assume $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ is any n-node undirected graph. The graph is *connected* if for each pair of distinct nodes i and j, there exists a path in $\mathcal{G}$ from i to j. A graph is said to be *disconnected* if it is not connected. A *component* of the graph $\mathcal{G}$ is a connected subgraph $\mathcal{G}'$ of $\mathcal{G}$ for which the addition of any edge in $\mathcal{E} - \mathcal{E}'$ to $\mathcal{G}'$ results in

a disconnected subgraph of $\mathcal{G}$. Thus, a disconnected graph can be viewed as the union of *at least* two components.

We can now define the important class of undirected graphs called trees: any undirected graph consisting of a single component with no cycles is a *tree*. The union of a collection of trees, assuming each tree's vertex set is disjoint from those of all others, is called a *forest*. A *subtree* of an undirected graph $\mathcal{G}$ is any subgraph of $\mathcal{G}$ that is itself a tree. A *spanning tree* of $\mathcal{G}$ is a subtree with vertex set equal to all of $\mathcal{V}$. Note that a path of an undirected graph can be viewed as a special case of a tree, in which context it is sometimes called a *chain*.

Given any pair of nodes i and j in a connected graph $\mathcal{G}$, their *distance* is the length of the minimum-length path among all paths from i to j in $\mathcal{G}$. A *clique* in $\mathcal{G}$ is any subset of nodes for which all pairs are connected by an edge in $\mathcal{G}$. Note that the set of all cliques in $\mathcal{G}$ trivially includes every one-node subset of $\mathcal{V}$ as well as the two-node subsets implied by the edges in $\mathcal{E}$. Indeed, these one-node and two-node subsets comprise the set of all cliques only if $\mathcal{G}$ is a tree (or a forest).

A tree also has the important property that its node set can always be *partially-ordered*; specifically, given $\mathcal{G}$ is a tree, we can always organize the node set $\mathcal{V}$ hierarchically in *scale* $s = 0, 1, 2, \ldots$ as follows. First choose an arbitrary *root* $i \in \mathcal{V}$ and assign it to scale zero; then, assign each other node $j \in \mathcal{V}\backslash i$ to the scale equal to its distance from i in $\mathcal{G}$. No matter the choice of root i, the sequence of node subsets associated with increasing scale yields a well-defined partial-ordering of $\mathcal{V}$ in $\mathcal{G}$.

Figure 2.1 illustrates an undirected graph along with examples of this terminology.

Directed Graphs

Assume $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ is any n-node directed graph. The *neighbors* of node i in $\mathcal{G}$ can be partitioned into the *parents* and the *children* of node i, denoted by subsets

$$pa(i) = \{j \in \mathcal{V} \mid (j, i) \in \mathcal{E}\} \quad \text{and} \quad ch(i) = \{j \in \mathcal{V} \mid (i, j) \in \mathcal{E}\},$$

respectively. The *in-degree* and *out-degree* of node i in $\mathcal{G}$ are the number of parents $|pa(i)|$ and the number of children $|ch(i)|$, respectively. A node i for which $pa(i) = \emptyset$ or $ch(i) = \emptyset$ is said to be a *parentless* or *childless* node in $\mathcal{G}$, respectively.

The *ancestors* of node i collectively refer to the parents $pa(i)$, the parents $pa(j)$ of each such parent $j \in pa(i)$, and so on ad infinitum, while always excluding i. Formally, initializing $\mathcal{V}_1(i) := pa(i)$ and applying the recursion

$$\mathcal{V}_{t+1}(i) := \bigcup_{j \in \mathcal{V}_t(i)} pa(j)\backslash i, \quad t = 1, 2, \ldots,$$

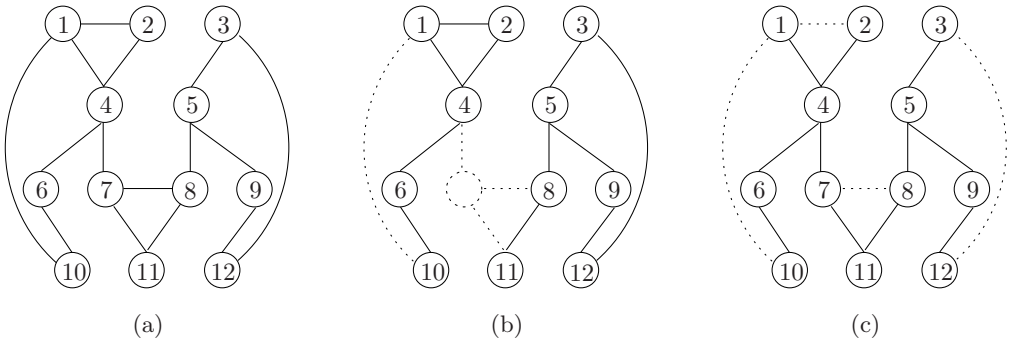


Figure 2.1. (a) A 12-node undirected graph $\mathcal{G}$ with four cycles, two of length four and the others of length three. We say that node 4 has neighbors $\{1, 2, 6, 7\}$, while node 5 has neighbors $\{3, 8, 9\}$. The set of all cliques in $\mathcal{G}$ consists of the one-node sets $\{\{i\}; i \in \mathcal{V}\}$, the two-node sets $\{\{i, j\} \in \mathcal{E}\}$ as well as the three-node sets $\{1, 2, 4\}$ and $\{7, 8, 11\}$. (b) A disconnected subgraph of $\mathcal{G}$ with two components, each with one cycle. (c) A spanning tree of $\mathcal{G}$, where one valid partial-order is $\{7\}$, $\{4, 11\}$, $\{1, 2, 6, 8\}$, $\{5, 10\}$, $\{3, 9\}$, $\{12\}$.

the set of ancestors of node i is the union $an(i) = \bigcup_t \mathcal{V}_t(i)$. The *descendants* $de(i)$ of node i are defined by the same union but where the recursion is initialized to the children $ch(i)$, then includes the children $ch(j)$ of each such child $j \in ch(i)$, and so on.

Given any directed graph $\mathcal{G}$, the graph obtained by substituting all edges in $\mathcal{G}$ by their undirected counterparts is called the *undirected topology* of $\mathcal{G}$. A directed graph is said to be *acyclic* if it contains no (directed) cycles, but note that its undirected topology need not necessarily be free of cycles. The special case in which the undirected topology of $\mathcal{G}$ is cycle-free, or a forest as defined above, is called a *polytree*.

A directed acyclic graph $\mathcal{G}$ has a number of important properties. Firstly, given any nodes i and j both in $\mathcal{V}$, the edge set cannot contain both (i, j) and (j, i). Secondly, for every node, the ancestors are disjoint from the descendants i.e., the intersection $an(i) \bigcap de(i)$ is empty for every $i \in \mathcal{V}$. Thirdly, the node set $\mathcal{V}$ can always be partially-ordered by *level* $\ell = 0, 1, \ldots$ as follows. Start by assigning all parentless nodes to level zero; then, assign each remaining node i to level ℓ only upon all of its parents $pa(i)$ being contained in the union of the node subsets associated with previous levels 0 to $\ell - 1$. The sequence of node subsets with increasing level yields the *forward partial-order* of $\mathcal{V}$ implied by $\mathcal{G}$. The analogous recursive construction, but based on children instead of parents, yields the *backward partial-order* of $\mathcal{V}$ implied by $\mathcal{G}$. A property unique to the special case of a polytree is that, for every node, no two parents share a common ancestor and, equivalently, no two children share a common descendant.

Figure 2.2 illustrates a directed graph along with examples of this terminology.

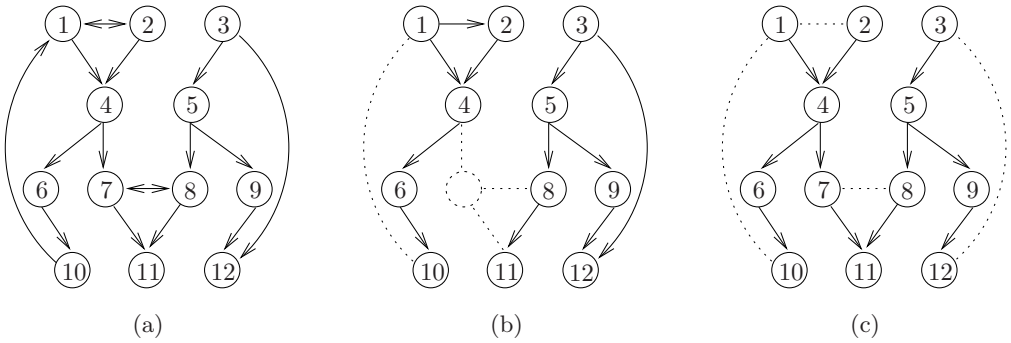


Figure 2.2. (a) A 12-node directed graph $\mathcal{G}$ with three cycles, one of length four and the others of length two. We say that node 4 has parents $\{1, 2\}$, children $\{6, 7\}$, ancestors $\{1, 2, 6, 10\}$ and descendants $\{1, 2, 6, 7, 8, 10, 11\}$; meanwhile, node 5 has parents $\{3\}$, children $\{8, 9\}$, ancestors $\{3\}$ and descendants $\{7, 8, 9, 11, 12\}$. (b) A directed acyclic subgraph of $\mathcal{G}$, where the forward partial order is $\{1, 3\}$, $\{2, 5\}$, $\{4, 8, 9\}$, $\{6, 11, 12\}$, $\{10\}$ and the backward partial-order is $\{10, 11, 12\}$, $\{6, 8, 9\}$, $\{4, 5\}$, $\{2, 3\}$, $\{1\}$. (c) A polytree subgraph of $\mathcal{G}$. Their respective undirected counterparts are shown in Figure 2.1.

■ 2.1.2 Elements of Probability Theory

A discrete (continuous) *random variable* X is defined by a discrete (Euclidean) set $\mathcal{X}$ and a probability mass (density) function $p_X : \mathcal{X} \to [0, \infty)$ for which the sum (integral) over $x \in \mathcal{X}$ evaluates to unity. We say that X takes values $x \in \mathcal{X}$ according to the *probability distribution* $p_X(x)$. In definitions that apply to either type of random variable, our convention is to assume X is discrete, understanding that any summation over values in $\mathcal{X}$ is replaced by the analogous integration if X is continuous. Any real-valued function of the form $c_X : \mathcal{X} \to \mathbb{R}$ will be called a *cost function* for X, and we say a cost $c_X(x)$ is assigned to each $x \in \mathcal{X}$. The subscript notation will be suppressed when the random variable involved is implied by the functional argument; that is, we let $p(x) \equiv p_X(x)$ and $c(x) \equiv c_X(x)$ for every x in $\mathcal{X}$. Also note that $p(X)$ and $c(X)$ are themselves well-defined random variables, each taking values in $\mathbb{R}$ according to a distribution derived from X and the functions p_X and c_X, respectively.

Let $X_1, X_2, \ldots, X_n$ denote n distinct random variables with *marginal* distributions $p(x_1), p(x_2), \ldots, p(x_n)$, respectively. The *random vector* $X = (X_1, \ldots, X_n)$ takes its values in the product set $\mathcal{X} = \mathcal{X}_1 \times \cdots \times \mathcal{X}_n$ according to a *joint* probability distribution $p(x) = p(x_1, \ldots, x_n)$. Consider any i and $x_i \in \mathcal{X}_i$ such that $p(x_i)$ is nonzero: assuming that $X_i = x_i$, let the vector of all other random variables $X_{\setminus i}$ take its values $x_{\setminus i} \in \mathcal{X}_{\setminus i}$ according to a probability distribution $p(x_{\setminus i}|x_i)$. The resulting function $p_{X_{\setminus i}|X_i} : \mathcal{X} \to [0, \infty)$, in essence a collection of up to $|\mathcal{X}_i|$ different distributions for random vector $X_{\setminus i}$, is called the *conditional* distribution of $X_{\setminus i}$ given X_i. The marginal, joint and

conditional distributions are related via the identities

$$p(x_i) = \sum_{x_{\setminus i} \in \mathcal{X}_{\setminus i}} p(x) \qquad \text{and} \qquad p(x) = p(x_i)p(x_{\setminus i}|x_i), \qquad i \in \{1, 2, \ldots, n\}.$$

The components of a random vector X are said to be *mutually independent* if their joint distribution satisfies

$$p(x) = \prod_{i=1}^{n} p(x_i) \qquad \Rightarrow \qquad p(x_{\setminus i}|x_i) = p(x_{\setminus i}), \quad i \in \{1, 2, \ldots, n\}.$$

Any two components X_i and X_j are said to be *independent* if $p(x_i, x_j) = p(x_i)p(x_j)$. Note that mutual independence is (in general) a stronger condition than independence between all $\binom{n}{2}$ pairs of component random variables.

The *expected cost* $\mathbf{E}[c(X)]$ denotes the sum (integral) over $x \in \mathcal{X}$ of the product $c(x)p(x)$. An important special case is the probability that X takes values in a given subset $\mathcal{A} \subset \mathcal{X}$, denoted by $\mathbf{P}[X \in \mathcal{A}]$, obtained by choosing $c(x)$ to be the indicator function on $\mathcal{A}$ i.e., unit cost for $x \in \mathcal{A}$ and zero cost otherwise. For a scalar random variable X, special cases include the *mean* and *variance* of X obtained by choosing $c(x) = x$ and $c(x) = (x - \mathbf{E}[X])^2$, respectively. In the case of a random vector, we distinguish between the *joint* expected cost $\mathbf{E}[c(X)] = \mathbf{E}[c(X_1, \ldots, X_n)]$ and a *conditional* expected cost $\mathbf{E}[c(X)|\, X_i = x_i]$, denoting the sum (integral) over $x_{\setminus i} \in \mathcal{X}_{\setminus i}$ of the product $c(x)p(x_{\setminus i}|x_i)$. The latter can be viewed as a particular cost function for X_i—indeed, evaluating its expected cost recovers the joint expected cost,

$$\mathbf{E}[\mathbf{E}[c(X)|\, X_i]] = \sum_{x_i \in \mathcal{X}_i} \mathbf{E}\left[c\left(X_{\setminus i}, x_i\right)\middle|\, X_i = x_i\right] p(x_i) = \mathbf{E}[c(X)], \qquad i \in \{1, 2, \ldots, n\}.$$

For a random vector X, its *mean vector* contains all of the component means, the ith element of which equals $\mathbf{E}[X_i]$. Given two component random variables X_i and X_j, their *covariance* is the joint expectation $\mathbf{E}[c(X_i, X_j)]$ with $c(x_i, x_j) = (x_i - \mathbf{E}[X_i])(x_j - \mathbf{E}[X_j])$, which specializes to the variance of X_i if $i = j$. We say X_i and X_j are *uncorrelated* if their covariance is zero, or equivalently that $\mathbf{E}[X_i X_j] = \mathbf{E}[X_i]\,\mathbf{E}[X_j]$, which is (in general) a weaker condition than independence. The *covariance matrix* of X contains all such pairwise covariances, each (i, j)th element equal to the covariance of X_i and X_j. Algebraically, this matrix is symmetric and positive semi-definite [93], and it is diagonal if the components of X are *mutually uncorrelated*, meaning every pair of distinct component variables are uncorrelated.

Example 2.1 (Gaussian Random Variables). A random variable X is said to be *Gaussian* (or normal) if its distribution takes the form

$$p(x) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right), \quad x \in \mathbb{R},$$

where μ and σ^2 denote, respectively, the mean and variance of X. This distribution has no analytical anti-derivative and, hence, calculating various probabilities for a Gaussian random variable is accomplished numerically. Assume the probability that a zero-mean, unit-variance Gaussian random variable W takes values less than or equal to w, denoted by

$$\phi(w) = \mathbf{P}\left[W \leq w\right] = \int_{-\infty}^{w} \frac{1}{\sqrt{2\pi}} \exp\left(-z^2/2\right)\, dz,$$

is available (as tabulated in e.g., [8]). Then, the probability that X takes values less than or equal to x is given by

$$\mathbf{P}\left[X \leq x\right] = \mathbf{P}\left[\frac{X-\mu}{\sigma} \leq \frac{x-\mu}{\sigma}\right] = \mathbf{P}\left[Y \leq \frac{x-\mu}{\sigma}\right] = \Phi\left(\frac{x-\mu}{\sigma}\right).$$

Example 2.2 (Gaussian Random Vectors). A length-n random vector X is said to be *multivariate Gaussian* if its distribution takes the form

$$p(x) = \frac{1}{(2\pi)^{n/2}|\Sigma|^{1/2}} \exp\left(-\frac{1}{2}(x-\mu)'\Sigma^{-1}(x-\mu)\right), \quad x \in \mathbb{R}^n,$$

where $\mu \in \mathbb{R}^n$ is the mean vector of X and $\Sigma \in \mathbb{R}^{n\times n}$ is the covariance matrix of X, while A', A^{-1} and $|A|$ denote the transpose, inverse and determinant, respectively, of any given matrix A [93]. Important properties include that (i) marginals and conditionals of a multivariate Gaussian distribution, as well as any linear transformation of a Gaussian random vector, all remain Gaussian and (ii) any two component random variables that are uncorrelated are also statistically independent [8], implying every Gaussian random vector with diagonal covariance matrix is a collection of mutually *independent* random variables.

■ 2.2 Bayesian Detection Models

Classical m-ary detection, or hypothesis testing, is perhaps the most elementary decision problem under uncertainty [108]. The basic setup is depicted in Figure 2.3. The true state $x \in \mathcal{X} = \{1, \ldots, m\}$ of the environment, taking one of $m \geq 2$ discrete values, is not fully observable but a sensor, upon receiving a noisy measurement $y \in \mathcal{Y}$, must generate a state-related decision $\hat{x} \in \mathcal{X}$ without delay. Subject to design is the *decision rule*, or the function $\gamma : \mathcal{Y} \to \mathcal{X}$ by which any particular measurement y is mapped to a particular decision $\hat{x}$. The choice of decision rule clearly depends on both a probabilistic model of the uncertain environment and the criterion by which one quantifies the rule-dependent decision performance.

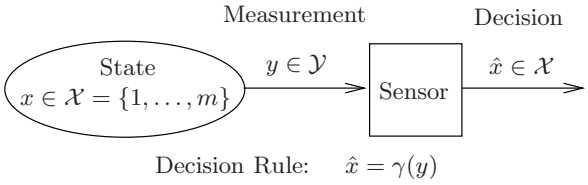


Figure 2.3. The single-sensor m-ary detection problem: a sensor receives noisy measurements from its otherwise unobservable, discrete-state environment, and subject to design is the rule by which the sensor generates its (discrete) state-related decision based on each received measurement.

■ 2.2.1 Minimum-Bayes-Risk Decision Criterion

Starting from the basic setup in Figure 2.3, the minimum-Bayes-risk criterion (i) assumes the (hidden) state process X and (observed) measurement process Y are jointly described by a given probability distribution $p(x, y)$, and (ii) assigns a numerical cost $c(\hat{x}, x)$ to each possible state-decision outcome. The performance of any rule-induced decision process $\hat{X} = \gamma(Y)$ is then measured by the expected cost, or *Bayes risk*,

$$J_d(\gamma) = \mathbf{E}\left[c(\hat{X}, X)\right] = \mathbf{E}\left[\mathbf{E}\left[c(\gamma(Y), X)|\, Y\right]\right] = \mathbf{E}\left[\sum_{x \in \mathcal{X}} c(\gamma(Y), x) p(x|Y)\right]. \tag{2.1}$$

An important special case of (2.1) is the error probability, which corresponds to choosing $c(\hat{x}, x)$ to be the indicator function on $\{(\hat{x}, x) \in \mathcal{X} \times \mathcal{X} \mid \hat{x} \neq x\}$.

Recognizing that $p(x|y) = p(x, y)/p(y)$ is proportional to $p(x)p(y|x)$ for every $y \in \mathcal{Y}$ such that $p(y)$ is nonzero, it follows that $\bar{\gamma}$ minimizes (2.1) if and only if

$$\bar{\gamma}(Y) = \arg\min_{\hat{x} \in \mathcal{X}} \sum_{x \in \mathcal{X}} c(\hat{x}, x) p(x) p(Y|x) \tag{2.2}$$

with probability one. Note that (i) the *likelihood function* $p(Y|x)$, taking its values $L(y) = \big(p_{Y|X}(y|1), \ldots, p_{Y|X}(y|m)\big)$ in the product set $\mathcal{L} = [0, \infty)^m \subset \mathbb{R}^m$, provides a sufficient statistic of the *online* measurement process Y and (ii) the parameter matrix $\theta \in \mathbb{R}^{m \times m}$, where the optimal values are given by

$$\bar{\theta}(\hat{x}, x) = p(x) c(\hat{x}, x),$$

can be specified *offline*, or prior to the processing of any actual measurements. In other words, given any particular measurement $Y = y$, implementation of (2.2) reduces to a matrix multiplication $\bar{\theta} L(y)$, yielding a length-m real-valued vector, followed by $m - 1$ comparisons to select the minimizing argument over $\hat{x} \in \mathcal{X}$.

One may also view the optimal detector in (2.2) as a particular partition of the likelihood set $\mathcal{L}$ into the regions $\mathcal{L}^1, \ldots, \mathcal{L}^m$, always choosing the decision $\hat{x}$ such that $L(y) \in \mathcal{L}^{\hat{x}}$. To see this, note that (2.2) implies

$$p(\hat{x}|y;\bar{\gamma}) = \begin{cases} 1 & , \quad \text{if } \hat{x} = \bar{\gamma}(y) \\ 0 & , \quad \text{otherwise} \end{cases}$$

and, in turn, the identity

$$p(\hat{x}|x;\bar{\gamma}) = \int_{y\in\mathcal{Y}} p(y|x)p(\hat{x}|y;\bar{\gamma})\,dy = \int_{y\in\{y'\in\mathcal{Y}|L(y')\in\mathcal{L}^{\hat{x}}\}} p(y|x)\,dy. \tag{2.3}$$

While the system of linear inequalities implied by (2.2) guarantees each region $\mathcal{L}^{\hat{x}} \subset \mathcal{L}$ is polyhedral, the associated subset of $\mathcal{Y}$ (via inversion of the likelihood function) may be non-polyhedral or disconnected, making the computation of $p(\hat{x}|x;\bar{\gamma})$ cumbersome if m grows large or the measurement model is complicated. Nonetheless, a characterization of $p(\hat{x}|x;\bar{\gamma})$ is essential to determine the achieved penalty

$$J_d(\bar{\gamma}) = \sum_{x\in\mathcal{X}} p(x) \sum_{\hat{x}\in\mathcal{X}} c(\hat{x},x)p(\hat{x}|x;\bar{\gamma}).$$

In problems where the integrals in (2.3) do not admit analytical solution nor reliable numerical approximation, Monte-Carlo methods (i.e., drawing samples from the joint process (X,Y) and simulating the decision process $\hat{X} = \bar{\gamma}(Y)$) can be employed to approximate empirically the distribution $p(\hat{x}|x;\bar{\gamma})$ or the expected cost $J_d(\bar{\gamma})$.

Example 2.3 (Binary Detectors, $m = 2$). The special case in which the hidden state process X takes just two values, which in preparation for future examples we label as -1 and $+1$, reduces the optimal rule in (2.2) to a particularly convenient form. Making the natural assumption that an error event $\hat{X} \neq x$ is more costly than an error-free event $\hat{X} = x$ for either possible value of x, (2.2) is equivalent to the *binary threshold rule*

$$\frac{p_{Y|X}(y|+1)}{p_{Y|X}(y|-1)} \equiv \Lambda(y) \quad \begin{array}{c} \hat{x}=+1 \\ > \\ < \\ \hat{x}=-1 \end{array} \quad \bar{\eta} \equiv \frac{\bar{\theta}(+1,-1)-\bar{\theta}(-1,-1)}{\bar{\theta}(-1,+1)-\bar{\theta}(+1,+1)}.$$

Here, the (scalar) quantity $\Lambda(y)$ is called the *likelihood-ratio* and $\bar{\eta}$ denotes the optimal value of a parameter $\eta \in [0,\infty)$ called the *threshold*. The error probability $\mathbf{P}\left[\hat{X} \neq X\right]$ is minimized by choosing $\eta = p_X(-1)/p_X(+1)$, and this threshold becomes unity if $p(x)$ is also uniform. The distribution $p(\hat{x}|x;\gamma)$ induced upon fixing the threshold η can be specified in terms of the so-called *false-alarm* and *detection* probabilities, denoted by $P^F(\eta)$ and $P^D(\eta)$, respectively, and defined by

$$\mathbf{P}\left[\hat{X}=+1\middle|\, X=x\right] = \mathbf{P}\left[\Lambda(Y) > \eta|\, X=x\right] = \begin{cases} P^F(\eta) & , \quad x=-1 \\ P^D(\eta) & , \quad x=+1 \end{cases}.$$

The subset of the unit plane $[0,1]^2$ defined by $\{(P^F(\eta), P^D(\eta)) \mid \eta \geq 0\}$ is called the *Receiver-Operating-Characteristic* curve—its many interesting properties (see e.g., [108]) are not needed in the scope of this thesis.

Example 2.4 (Linear Binary Detectors). The special case of a linear-Gaussian measurement model allows the decision regions in measurement space $\mathcal{Y}$ to retain the polyhedral form of their counterparts in likelihood space $\mathcal{L}$, simplifying the multi-dimensional integrals that must be solved to obtain the rule-dependent distribution $p(\hat{x}|x;\gamma)$. Starting with the binary problem in Example 2.3, denote by μ^- and μ^+ the real-valued vector signals associated to the two possible states and assume the measurement process is

$$Y = \mu^X + W$$

where the additive noise process W is a zero-mean Gaussian random vector with (known) covariance matrix Σ (see Example 2.2). The resulting likelihood function $p(y|x)$ consists of a pair of multivariate Gaussian distributions with mean vectors μ^- and μ^+, respectively, and common covariance matrix Σ. In turn, the likelihood-ratio specializes to

$$\Lambda(y) = \exp\left(-\frac{1}{2}(y-\mu^+)'\Sigma^{-1}(y-\mu^+) + \frac{1}{2}(y-\mu^-)'\Sigma^{-1}(y-\mu^-)\right)$$

and the binary threshold rule as a function of η reduces to

$$(\mu^+ - \mu^-)'\Sigma^{-1}y \quad \begin{matrix} \hat{x}=+1 \\ > \\ < \\ \hat{x}=-1 \end{matrix} \quad \log(\eta) + \frac{1}{2}\left(\mu^{+\prime}\Sigma^{-1}\mu^+ - \mu^{-\prime}\Sigma^{-1}\mu^-\right),$$

a form that is linear in the measurement vector y. If the components of W are also mutually independent, so that Σ is diagonal with the (i,i)th element denoted by σ_i^2, then the rule is

$$\sum_{i=1}^n \left(\frac{\mu_i^+ - \mu_i^-}{\sigma_i^2}\right) y_i \quad \begin{matrix} \hat{x}=+1 \\ > \\ < \\ \hat{x}=-1 \end{matrix} \quad \log(\eta) + \frac{1}{2}\sum_{i=1}^n \frac{\mu_i^+\mu_i^+ - \mu_i^-\mu_i^-}{\sigma_i^2}.$$

If the components of Y are also identically distributed, meaning conditional means $\mu_i^\pm = \mu^\pm$ and variances $\sigma_i^2 = \sigma^2$ for all i, then

$$\sum_{i=1}^n y_i \quad \begin{matrix} \hat{x}=+1 \\ > \\ < \\ \hat{x}=-1 \end{matrix} \quad \left(\frac{\sigma^2}{\mu^+ - \mu^-}\right)\log(\eta) + \frac{n}{2}\left(\mu^- + \mu^+\right).$$

Note that this rule continues to require joint processing of the component measurements $y = (y_1, \ldots, y_n)$. Finally, if Y is also scalar, the binary threshold rule with parameter η (in likelihood space) can be implemented via a threshold rule in measurement space, comparing y to the threshold $\tau = \left(\frac{\sigma^2}{\mu^+ - \mu^-}\right)\log(\eta) + \frac{\mu^- + \mu^+}{2}$. Accordingly, the false-alarm and detection probabilities simplify to

$$\mathbf{P}\left[\Lambda(Y) > \eta \middle| X = x\right] = \mathbf{P}\left[Y > \tau \middle| X = x\right] = \int_\tau^\infty p(y|x)\,dy = 1 - \Phi\left(\frac{\tau - \mu^x}{\sigma}\right)$$

with function Φ as defined in Example 2.1.

Subsequent chapters of this thesis will rely on the model in Figure 2.3 but where the decision space, call it $\mathcal{U} = \{1, \dots, d\}$, can have cardinality d different from the state cardinality m. The structure of the problem is essentially unchanged. Decision rules and costs take the form of functions $\gamma : \mathcal{Y} \to \mathcal{U}$ and $c(u, x)$, respectively. It follows that the optimal detector in (2.2) uses rule parameters $\bar{\theta}(u, x) = p(x)c(u, x)$, performing $d - 1$ comparisons to select the minimizing argument over $u \in \mathcal{U}$. Similarly, (2.2) can be viewed as a particular partition of the likelihood set $\mathcal{L}$ into the regions $\mathcal{L}^1, \dots, \mathcal{L}^d$ so that the the rule-dependent distribution in (2.3) is

$$p(u|x; \bar{\gamma}) = \int_{y \in \mathcal{Y}} p(y|x) p(u|y; \bar{\gamma})\, dy = \int_{y \in \{y' \in \mathcal{Y} | L(y') \in \mathcal{L}^u\}} p(y|x)\, dy.$$

Example 2.5 (Binary Detectors with Non-Binary Decision Spaces). Consider the binary detector in Example 2.3 but with decision space $\mathcal{U} = \{1, \dots, d\}$ for $d \geq 2$. Any given rule parameters $\theta \in \mathbb{R}^{d \times m}$ define a particular partition of the likelihood-ratio space $[0, \infty)$ into (at most) d subintervals, characterized by $d - 1$ threshold values satisfying

$$0 \leq \eta^1 \leq \eta^2 \leq \cdots \leq \eta^{d-1} \leq \infty.$$

This *monotone threshold rule* alongside the natural assumption that the d elements of $\mathcal{U}$ are labeled such that

$$\frac{\mathbf{P}\left[U = u \middle| X = +1\right]}{\mathbf{P}\left[U = u \middle| X = -1\right]} \leq \frac{\mathbf{P}\left[U = u + 1 \middle| X = +1\right]}{\mathbf{P}\left[U = u + 1 \middle| X = -1\right]}, \quad u = 1, \dots, d - 1,$$

simplifies to making the decision $u \in \mathcal{U}$ such that $\Lambda(y) \in [\eta^{u-1}, \eta^u)$, taking $\eta^0 = 0$ and $\eta^d = \infty$. In the special case of a scalar linear binary detector (see Example 2.4), we retain the analogous partition in measurement space $(-\infty, \infty)$ with respective thresholds

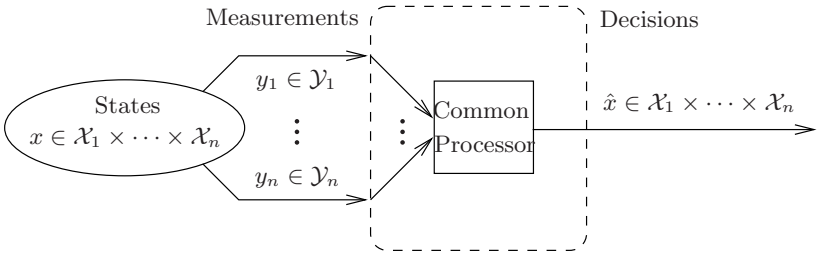
$$-\infty \leq \tau^1 \leq \tau^2 \leq \cdots \leq \tau^{d-1} \leq \infty$$

determined by $\tau^u = \left(\frac{\sigma^2}{\mu^+ - \mu^-}\right) \log(\eta^u) + \frac{\mu^- + \mu^+}{2}$. The rule is simplified to making decision u such that $y \in [\tau^{u-1}, \tau^u)$, implying

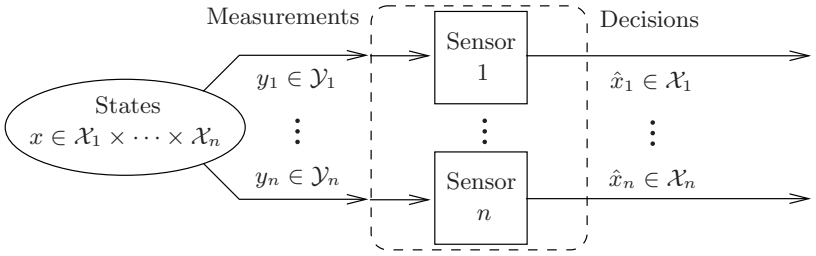
$$\mathbf{P}\left[U = u \middle| X = x\right] = \mathbf{P}\left[y \in [\tau^{u-1}, \tau^u) \middle| X = x\right] = \Phi\left(\frac{\tau^u - \mu^x}{\sigma}\right) - \Phi\left(\frac{\tau^{u-1} - \mu^x}{\sigma}\right).$$

■ 2.2.2 Baseline Multi-Sensor Decision Strategies

The generalization of (2.1)–(2.3) to n sensors, as depicted in Figure 2.4(a), is conceptually simple: let the states and measurements take values in product sets $\mathcal{X} = \mathcal{X}_1 \times \cdots \times \mathcal{X}_n$ and $\mathcal{Y} = \mathcal{Y}_1 \times \cdots \times \mathcal{Y}_n$, respectively, the components $x_i \in \mathcal{X}_i$ and $y_i \in \mathcal{Y}_i$ denoting the discrete state and noisy measurement of the environment local to the ith sensor. However, then m scales exponentially with n, so performing directly the computations implied by (2.1)–(2.3) becomes challenging for even modest values of n. The following two examples illustrate some of the computational challenges associated with optimal processing in the n-sensor model.



(a) Optimal Centralized Strategy: $(\hat{x}_1, \ldots, \hat{x}_n) = \bar{\gamma}(y_1, \ldots, y_n)$



(b) Myopic Decentralized Strategy: $\hat{x}_i = \bar{\delta}_i(y_i)$, $i = 1, \ldots, n$

Figure 2.4. The two baseline multi-sensor decision strategies for processing spatially-distributed measurements, (a) the optimal centralized strategy for which online communication overhead can be unbounded and (b) the myopic decentralized processing strategy for which online communication overhead is zero.

Example 2.6 (Maximum-A-Posterior (MAP) Estimation). In the n-sensor Bayesian detection model, assume the global cost function

$$c(\hat{x}, x) = \begin{cases} 1 & , \quad \text{if } \hat{x}_i \neq x_i \text{ for at least one component } i \\ 0 & , \quad \text{otherwise} \end{cases} .$$

The risk in (2.1) specializes to the error probability $\mathbf{P}\left[\hat{X} \neq X\right]$ and the optimal detector in (2.2) specializes to

$$\bar{\gamma}(Y) = \arg\max_{x \in \mathcal{X}} p(x|Y),$$

referred to as the *Maximum A-Posterior* (MAP) strategy. If the prior probabilities $p(x)$ are also uniform over $\mathcal{X}$, in which case $p(x|y) \propto p(y|x)$ for every $y \in \mathcal{Y}$ such that $p(y) > 0$, we obtain the *Maximum-Likelihood* (ML) strategy,

$$\bar{\gamma}(Y) = \arg\max_{x \in \mathcal{X}} p(Y|x).$$

Direct implementation of either strategy, per measurement $Y = y$, amounts to solving an integer program over a solution space that scales exponentially with n.

Example 2.7 (Maximum-Posterior-Marginal (MPM) Estimation). In the n-sensor Bayesian detection model, assume the global cost function

$$c(\hat{x},x)=\sum_{i=1}^{n}c(\hat{x}_i,x_i),\qquad \text{where}\quad c(\hat{x}_i,x_i)=\begin{cases}1 & , \text{ if } \hat{x}_i\neq x_i\\ 0 & , \text{ otherwise}\end{cases}\qquad \text{for } i=1,\dots,n.$$

The risk in (2.1) specializes to the sum-error probability, or the expected number of component errors between vectors $\hat{X}$ and X, and the optimal detector in (2.2) specializes to

$$\bar{\gamma}(Y)=(\bar{\gamma}_1(Y),\dots,\bar{\gamma}_n(Y)),\qquad \text{where}\quad \bar{\gamma}_i(Y)=\arg\max_{x_i\in\mathcal{X}_i}p(x_i|Y)\quad \text{for } i=1,\dots,n,$$

referred to as the *Maximum-Posterior-Marginal* (MPM) strategy. This strategy is easy to implement given the local marginals $p(x_i|y)$ conditioned on all measurements $Y=y$; of course, starting from the global posterior $p(x|y)$, direct computation of each ith such local posterior involves summation over a number of terms that scales exponentially with n.

In the next section, we discuss graph-based message-passing algorithms that efficiently implement the n-sensor generalization of (2.2), in the sense that total computation overhead (per online decision) scales only linearly in n. However, notice that (2.2) pays no attention to the practical caveat that sensors may be arranged in a spatially-distributed network. That is, the n-sensor generalization of (2.2) is said to be a *centralized* processing strategy, where the requirement to evaluate the global likelihood vector $p(y|x)$ before making an optimal decision $\hat{x}=\bar{\gamma}(y)$ assumes all n measurements (or at least their sufficient statistics e.g., their weighted sum in Example 2.4, the posterior marginals $p(x_i|y)$ local to each node i in Example 2.7) have been reliably communicated via the network. We say that an n-sensor processing strategy is *decentralized* if it must make decisions based on strictly less information than is assumed to be available by the optimal centralized strategy (e.g., due to algorithmic resource constraints beyond those satisfied by even the most efficient centralized implementations).[1]

The remaining chapters of this thesis develop and analyze a particular class of decentralized strategies, assuming the dominant resource constraints arise from the unreliable (and costly) communication medium. The graph-based message-passing algorithms described in the next section will (precluding trivial problem instances) imply that the centralized communication overhead cannot be less than $n-1$ real-valued messages (per online decision). In contrast, we will assume a non-ideal communication model from the start, constraining communication overhead to no more than a fixed number of discrete-valued messages, or *symbols*, and, in turn, seeking the feasible strategy that best mitigates the potential *loss* from optimal centralized performance.

[1] This distinction between centralized and decentralized strategies precludes certain trivial instances of the multi-sensor problem formulation, namely those for which the optimal centralized strategy degenerates to a feasible decentralized strategy.

A trivial member in our class of decentralized strategies is the *myopic* strategy, having zero communication overhead; see Figure 2.4(b). It assumes each sensor i is initialized knowing only its *local* model for Bayesian detection i.e., the distribution $p(x_i, y_i)$ and a cost function $c(\hat{x}_i, x_i)$, and, using only this local model, determines its component estimate $\hat{x}_i$ from the local measurement y_i as if in isolation i.e., the rule at node i is

$$\bar{\delta}_i(Y_i) = \arg\min_{\hat{x}_i \in \mathcal{X}_i} \sum_{x_i \in \mathcal{X}_i} \underbrace{c(\hat{x}_i, x_i)p(x_i)}_{\bar{\phi}_i(\hat{x}_i, x_i) \in \mathbb{R}} p(Y_i|x_i). \tag{2.4}$$

That is, the myopic strategy is a particular collection of single-sensor decision rules $\bar{\delta} = (\bar{\delta}_1, \ldots, \bar{\delta}_n)$, specified offline by parameters $\bar{\phi} = (\bar{\phi}_1, \ldots, \bar{\phi}_n)$, where no one node transmits nor receives information and total online computation scales linearly with n.

It is easy to see that the myopic strategy is sub-optimal, meaning $J_d(\bar{\delta}) \geq J_d(\bar{\gamma})$ over all multi-sensor problem instances. Equality is achieved only in certain degenerate (and arguably uninteresting) cases, including the zero cost function i.e., $c(\hat{x}, x) = 0$ for all $(\hat{x}, x) \in \mathcal{X} \times \mathcal{X}$, or the case of n unrelated single-sensor problems i.e.,

$$p(x, y) = \prod_{i=1}^{n} p(x_i, y_i) \qquad \text{and} \qquad c(\hat{x}, x) = \sum_{i=1}^{n} c(\hat{x}_i, x_i).$$

More generally, the extent to which the myopic strategy $\bar{\delta}$ falls short from optimal centralized performance, or the loss $J_d(\bar{\delta}) - J_d(\bar{\gamma})$, remains a complicated function of the *global* detection model i.e., the distribution $p(x, y)$ and cost function $c(\hat{x}, x)$.

While the optimal centralized strategy $\bar{\gamma}$ and the myopic decentralized strategy $\bar{\delta}$ are both functions that map $\mathcal{Y}$ to $\mathcal{X}$, the different processing assumptions amount to different size-m partitions of the joint likelihood space $\mathcal{L}$. In particular, under myopic processing assumptions, the strategy-dependent conditional distribution in the integrand of (2.3) inherits the factored structure

$$p(\hat{x}|y; \bar{\delta}) = \prod_{i=1}^{n} p(\hat{x}_i|y_i; \bar{\delta}_i),$$

where each ith term involves variables only at the individual node i. This structure can lead to desirable computational ramifications: to illustrate, suppose only the decision at node i is costly, captured by choosing $c(\hat{x}, x) = c(\hat{x}_i, x_i)$ for all $(\hat{x}, x) \in \mathcal{X} \times \mathcal{X}$. Then, the strategy $\bar{\gamma}_i : \mathcal{Y} \to \mathcal{X}_i$ defined by selecting the ith component of $\hat{x} = \bar{\gamma}(y)$ is the

global minimizer of (2.1), achieving penalty

$$\begin{aligned} J_d(\bar{\gamma}_i) = \mathbf{E}\left[c(\hat{X}_i, X_i)\right] &= \mathbf{E}\left[\mathbf{E}\left[c(\bar{\gamma}_i(Y), X_i)\middle| Y\right]\right] \\ &= \sum_{x_i \in \mathcal{X}_i} p(x_i) \sum_{\hat{x}_i \in \mathcal{X}_i} c(\hat{x}_i, x_i) \underbrace{\sum_{x_{\setminus i}} \sum_{\hat{x}_{\setminus i}} p(x_{\setminus i}|x_i) p(\hat{x}|x; \bar{\gamma})}_{p(\hat{x}_i|x_i;\bar{\gamma}_i)} \end{aligned}$$

in contrast, the local myopic rule $\bar{\delta}_i$ minimizes (2.1) over only the subset of all such rules having the form $\delta_i : \mathcal{Y}_i \to \mathcal{X}_i$, achieving penalty

$$\begin{aligned} J_d(\bar{\delta}_i) &= \mathbf{E}\left[\mathbf{E}\left[c(\bar{\delta}_i(Y_i), X_i)\middle| Y_i\right]\right] \\ &= \sum_{x_i \in \mathcal{X}_i} p(x_i) \sum_{\hat{x}_i \in \mathcal{X}_i} c(\hat{x}_i, x_i) \underbrace{\int_{y_i \in \mathcal{Y}_i} p(y_i|x_i) p(\hat{x}_i|y_i; \bar{\delta}_i)\, dy_i}_{p(\hat{x}_i|x_i;\bar{\delta}_i)} \end{aligned} \tag{2.5}$$

regardless of the non-local conditional distribution $p(x_{\setminus i}, y_{\setminus i}|x_i, y_i)$ and the collective strategy $\delta_{\setminus i} : \mathcal{Y}_{\setminus i} \to \mathcal{X}_{\setminus i}$ of all other nodes. Thus, assuming myopic processing constraints and focusing on a cost function local to node i, the global penalty J_d involves sums and integrals over only local random variables $(X_i, Y_i, \hat{X}_i)$. This simplification foreshadows the key problem structure to be exploited in subsequent chapters of this thesis, seeking to retain a similarly tractable decomposition of the general n-sensor sums and integrals, yet also relaxing the constraint of zero online communication and considering costs that can depend on all sensors' decisions.

■ 2.3 Probabilistic Graphical Models

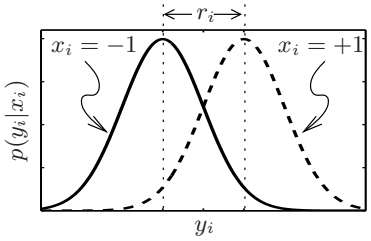
Many estimation problems, including those motivating this thesis (e.g., the n-sensor detection problems in Example 2.6 and Example 2.7), involve the joint distribution of a large number of random variables. The formalism of graphical models [28, 49, 51, 60, 79, 117, 120] provides both a compact representation of large random vectors and a systematic characterization of probabilistic structure to be exploited for computational efficiency. We focus on models in which, given an n-node (directed or undirected) graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, each node i in $\mathcal{V}$ is identified with a pair of random variables, a hidden (discrete) random variable X_i and an observable (discrete or continuous) random variable Y_i. The joint distribution of the respective random vectors $X = \{X_i; i \in \mathcal{V}\}$ and $Y = \{Y_i; i \in \mathcal{V}\}$ takes the form

$$p(x, y) = p(x) \prod_{i \in \mathcal{V}} p(y_i|x_i), \tag{2.6}$$

where prior probabilities $p(x)$ are represented as a set of local interactions among different subvectors of X in correspondence with the edge set $\mathcal{E}$ of $\mathcal{G}$. This representation encodes structure built upon a precise correspondence between the probabilistic concept of conditional independence and the graph-theoretic concept of node separation. Furthermore, these conditional independencies allow the computation of key summarizing statistics to be organized recursively, leading to especially efficient optimal algorithms (i.e., scaling linearly with $|\mathcal{V}| = n$) when the underlying graph $\mathcal{G}$ is tree-structured.

Example 2.8 (Linear Binary Detectors in Spatially-Uncorrelated Noise). Let there be n spatially-distributed sensors, each ith such sensor a scalar linear binary detector (see Example 2.4) with *local* likelihood function given by

$$p(y_i|x_i) = \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{1}{2}\left(y_i - \frac{x_i r_i}{2}\right)^2\right), \quad x_i \in \{-1, +1\} \text{ and } y_i \in \mathbb{R}.$$



Likelihood Function at Node i

Here, we have chosen state-related means $\pm\frac{r_i}{2}$ and a unit-variance noise process W_i so that the single parameter $r_i \in (0, \infty)$ captures the effective noise level e.g., measurements by sensor i are less noisy, or equivalently more informative on the average, than measurements by sensor j if $r_i > r_j$. If the Gaussian noise processes $W_1, \ldots, W_n$ are mutually uncorrelated (i.e., the case of a diagonal covariance matrix Σ in Example 2.4), then the observable processes $Y_1, \ldots, Y_n$ are mutually independent conditioned on the global hidden process $X = x$ i.e., the global likelihood function $p(y|x)$ satisfies (2.6).

■ 2.3.1 Compact Representations

There are two types of graphical models for a given random vector X, depending on whether the underlying graph $\mathcal{G}$ is undirected or directed. The respective edge sets make, in general, distinct assertions on the conditional independence properties satisfied by the joint distribution $p(x)$. For models in which the undirected topology of $\mathcal{G}$ is a tree (e.g., as in Example 2.9), these distinctions seem almost superfluous because the two representations are defined on this same tree topology. For non-tree-structured models, however, maintaining equivalence between the two types of representations requires more care. The following examples discuss the most commonly studied graphical models, introducing key concepts we treat more formally in the next two subsections.

Example 2.9 (Hidden Markov Models). An important special case of the joint distribution in (2.6) is called a *hidden Markov model* [14, 27, 84]. The prior probabilities $p(x)$ are described by a temporal *Markov chain* [8, 33], typically expressed as the product of an initial state distribution and so-called transition probabilities,

$$p(x) = p(x_1) \prod_{i=1}^{n-1} p(x_{i+1}|x_i).$$

Note that this distribution factors over pairs of random variables (X_i, X_{i+1}) in correspondence with directed edge set $\{(i, i+1); i = 1, \ldots, n-1\}$, or a length-$n$ path from node 1 to node n. An equivalent representation, using the identity $p(x_{i+1}|x_i) = p(x_i, x_{i+1})/p(x_i)$, is given by

$$p(x) = \prod_{i=1}^{n} p(x_i) \prod_{i=1}^{n-1} \frac{p(x_i, x_{i+1})}{p(x_i)p(x_{i+1})},$$

in which the factors more naturally correspond to the undirected counterpart of the underlying path. Another equivalent representation is

$$p(x) = p(x_n) \prod_{i=1}^{n-1} p(x_i|x_{i+1}),$$

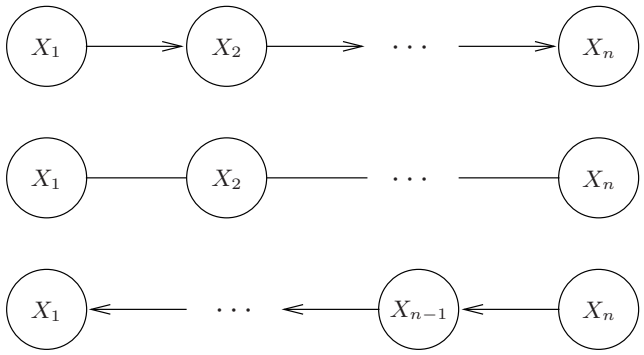
having factors in correspondence with the same path but in the reverse direction. Regardless of the particular representation, the fundamental property of a Markov chain is that the past and future are conditionally independent given the present i.e.,

$$p(x_{\setminus i}|x_i) = p(x_1, \ldots, x_{i-1}|x_i)p(x_{i+1}, \ldots, x_n|x_i), \quad i = 1, 2, \ldots n.$$

It follows from these conditional independence properties that

$$p(x_i|x_{\setminus i}) = \frac{p(x)}{p(x_{\setminus i})} = \frac{p(x_{i-1}, x_i, x_{i+1})}{p(x_{i-1}, x_{i+1})} = p(x_i|x_{i-1}, x_{i+1}), \quad i = 1, 2, \ldots, n,$$

or that each hidden variable X_i is conditionally independent of all non-neighboring variables given both (X_{i-1}, X_{i+1}). The respective graphical models for these three mathematically equivalent representations of an n-step Markov chain model are illustrated below.



Example 2.10 (Multiresolution Markov Models). Another important special case of the joint distribution in (2.6) is a *multiresolution Markov model* [20, 59, 61, 120], essentially generalizing the properties described in Example 2.9 for a simple chain to any underlying

graph whose undirected topology is cycle-free. One representation of a tree-structured random process X is as a Markov chain indexed in *scale*, starting from any particular root node $sc(0) \in \mathcal{V}$ and categorizing the remaining nodes into disjoint subsets $\{sc(s); s = 1, 2, \dots d\}$ for some $d \leq n-1$ according to their distance s from this root i.e.,

$$p(x) = p(x_{sc(0)}) \prod_{s=1}^{d} p(x_{sc(s)}|x_{sc(s-1)}).$$

Moreover, these transition probabilities further decompose within each scale, viewing the unique path from root to any other node as its own temporal Markov chain: letting $pa(i)$ denote the node in scale $s-1$ that lies on the path from root to the node i in scale s,

$$p(x_{sc(s)}|x_{sc(s-1)}) = \prod_{i \in sc(s)} p(x_i|x_{pa(i)}).$$

The distribution $p(x)$ thus factors in correspondence with directed edges $\mathcal{E} = \{(pa(i), i); i \in \mathcal{V}\backslash sc(0)\}$. Recognizing that $\bigcup_{s=1}^{d} sc(s) = \mathcal{V}\backslash sc(0)$ and employing the identity $p(x_i|x_{pa(i)}) = p(x_{pa(i)}, x_i)/p(x_{pa(i)})$, we may equivalently write

$$p(x) = \prod_{i \in \mathcal{V}} p(x_i) \prod_{(i,j) \in \mathcal{E}} \frac{p(x_i, x_j)}{p(x_i)p(x_j)}, \tag{2.7}$$

removing the asymmetry that resulted from the arbitrary choice of root node. Figure 2.5(a) illustrates the graphical models implied by these equivalent representations.

Example 2.11 (Nearest-Neighbor Grid Models). Yet another important special case of the joint distribution in (2.6) is a *nearest-neighbor grid model*, which is distinct from the previous two examples in that the underlying graph has cycles. As shown in Figure 2.5(b), each hidden variable inside the grid's perimeter is connected to its four closest spatial neighbors, while corner variables have two neighbors and all other variables on the perimeter have three neighbors. Analogous to the conditional independencies noted for the Markov chain in Example 2.9, the implication here is that each hidden variable X_i is conditionally independent of all non-neighboring processes given only its neighboring processes i.e.,

$$p(x_i|x_{\backslash i}) = p(x_i|x_{ne(i)}), \quad i = 1, 2, \dots n.$$

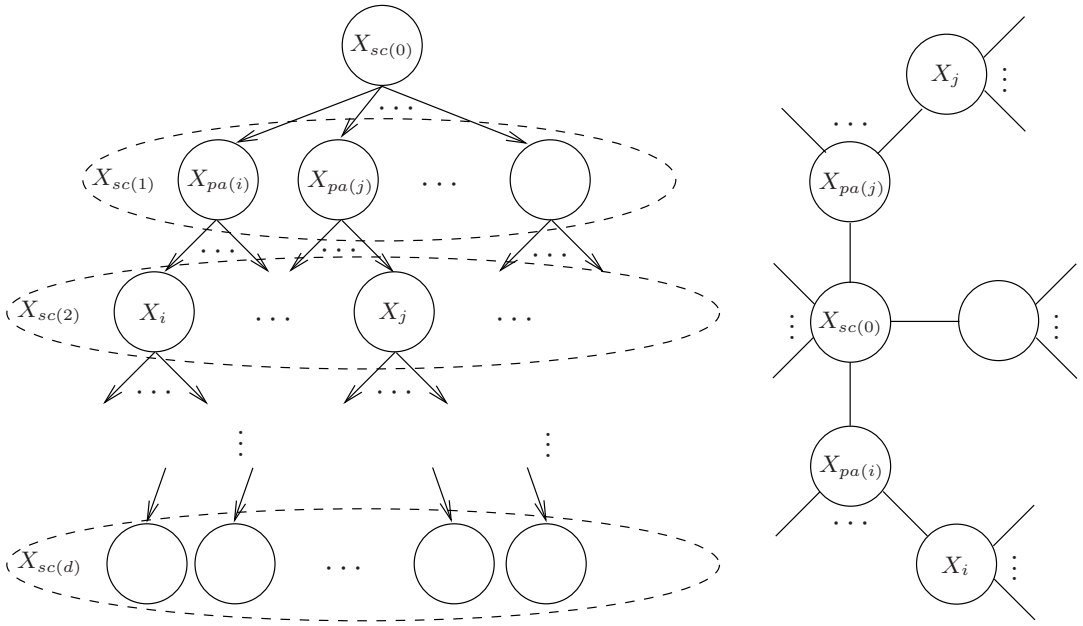
As will be described more formally in the next subsection, this graphical model encompasses all prior distributions having the structural form (up to normalization)

$$p(x) \propto \prod_{(i,j) \in \mathcal{E}} \psi_{i,j}(x_i, x_j),$$

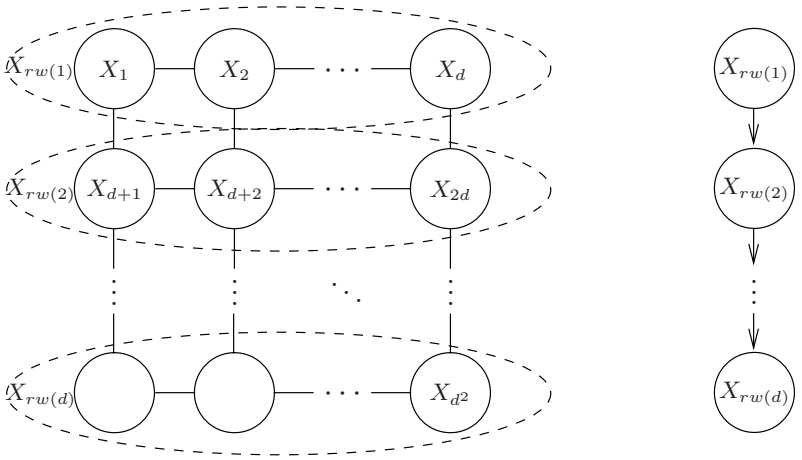
where $\psi_{i,j} : \mathcal{X}_i \times \mathcal{X}_j \to [0, \infty)$ denotes any nonnegative real-valued function of the variables connected by each edge (i, j) in $\mathcal{E}$. Note that, in contrast to the tree-structured models in Example 2.9 and Example 2.10, each factor $\psi_{i,j}$ need not be a valid (joint or conditional) probability distribution for X_i and X_j. Yet if we aggregate the variables row-by-row, defining "super-nodes" for the subvectors $X_{rw(s)} = (X_{ds-d+1}, \dots, X_{ds})$ for $s = 1, 2, \dots, d$, then we may write

$$p(x_{rw(s)}|x_{\backslash rw(s)}) = p(x_{rw(s)}|x_{rw(s-1)}, x_{rw(s+1)}),$$

which corresponds to a Markov chain on these aggregated variables.



(a) Two Equivalent Representations of a d-Scale Multiresolution Markov Model



(b) Two Equivalent Representations of a d-by-d Nearest-Neighbor Grid Model

Figure 2.5. Well-studied examples of directed and undirected graphical models for the compact representation of prior probabilities $p(x)$ in large-scale estimation problems (see Examples 2.9–2.11). The tree-structured models in (a) admit representation using either type of graph without modification to its undirected topology, whereas the lack of a natural partial-order in graphs with cycles prohibits a valid directed representation for (b) without first introducing node aggregation and losing explicit structure, illustrated here by the Markov chain on the row-by-row subvectors of the original grid model.

Undirected Graphical Models (Markov Random Fields)

An undirected graphical model, or *Markov random field*, rests upon an undirected graph $\mathcal{G}$. We say that the random vector X is (globally) *Markov* with respect to the graph $\mathcal{G}$ if, whenever node set $\mathcal{V}_2$ separates the node sets $\mathcal{V}_1$ and $\mathcal{V}_3$, the subvectors $X_{\mathcal{V}_1}$ and $X_{\mathcal{V}_3}$ are independent conditioned on $X_{\mathcal{V}_2}$ i.e.,

$$p(x_{\mathcal{V}_1}, x_{\mathcal{V}_3}|x_{\mathcal{V}_2}) = p(x_{\mathcal{V}_1}|x_{\mathcal{V}_2})p(x_{\mathcal{V}_3}|x_{\mathcal{V}_2}). \tag{2.8}$$

As an example, the model with underlying graph shown in Figure 2.1(a) implies the conditional independencies

$$p(x_1, x_2, x_3, x_4, x_5|x_7, x_8) = p(x_1, x_2, x_3|x_7, x_8)p(x_4, x_5|x_7, x_8),$$

$$p(x_5, x_{12}|x_3, x_9) = p(x_5|x_3, x_9)p(x_{12}|x_3, x_9)$$

and many others. Important special cases of (2.8) are the (local) Markov properties

$$p(x_i|x_{\mathcal{V}\setminus i}) = p(x_i|x_{ne(i)}), \quad i = 1, \ldots, n,$$

stating that each X_i, conditioned on the immediate neighbors' hidden variables $X_{ne(i)}$, is independent of all other hidden variables in the model.

In a general graph $\mathcal{G}$, the connection between the set of all Markov properties and a joint distribution satisfying them is not as readily apparent as was the case in Example 2.9. The celebrated *Hammersley-Clifford* theorem [13, 35] provides a sufficient condition (also necessary if $p(x)$ is strictly positive for all $x \in \mathcal{X}$): denoting by $\mathbf{C}$ the collection of all cliques $\mathcal{C} \subset \mathcal{V}$ in $\mathcal{G}$, the random vector X is Markov with respect to $\mathcal{G}$ if (and only if for strictly positive distributions)

$$p(x) \propto \prod_{\mathcal{C}\in\mathbf{C}} \psi_{\mathcal{C}}(x_{\mathcal{C}}), \tag{2.9}$$

where each *clique potential* $\psi_{\mathcal{C}}$ represents some nonnegative real-valued function of its arguments.[2] It is easily seen that it suffices to restrict the collection $\mathbf{C}$ to only the set of *maximal cliques* in $\mathcal{G}$, or only the cliques that are *not* a strict subset of any other clique. It is typically *not* the case that the right-hand-side of (2.9) sums to unity, so to achieve equality with $p(x)$ requires normalizing the right-hand-side, in general a sum over a number of terms exponential in n. It is also not necessarily the case, even if

[2] A clique potential is conventionally the quantity $\log \psi_{\mathcal{C}}$, calling $\psi_{\mathcal{C}}$ a *compatibility function*, but this distinction can be ignored in the scope of this thesis.

we assume the right-hand-side is normalized, that any particular clique potential $\psi_{\mathcal{C}}$ is itself a valid probability distribution for the subvector $X_{\mathcal{C}}$.

The graph-dependent structure exhibited by (2.9) defines a family of probability distributions for random vector X, members of which correspond to different choices of clique potentials. The graph structure can also be viewed as placing explicit constraints on the set of all valid distributions for X. This structure can often be further constrained by the choice of clique potentials: for example, assuming each clique potential factors in direct correspondence with the edges in the respective clique, (2.9) specializes to the *pairwise* representation

$$p(x) \propto \prod_{i \in \mathcal{V}} \psi_i(x_i) \prod_{(i,j) \in \mathcal{E}} \psi_{i,j}(x_i, x_j), \tag{2.10}$$

where (nonnegative real-valued) functions ψ_i and $\psi_{i,j}$ are called *node potentials* and *edge potentials*, respectively.[3] Note that all models discussed in Examples 2.9–2.11 admit representation of the form in (2.10); moreover, in the special case that $\mathcal{G}$ is a tree, we achieve equality in (2.10) by choosing $\psi_i(x_i) = p(x_i)$ and $\psi_{i,j}(x_i, x_j) = \frac{p(x_i,x_j)}{p(x_i)p(x_j)}$.

Directed Graphical Models (Bayesian Networks)

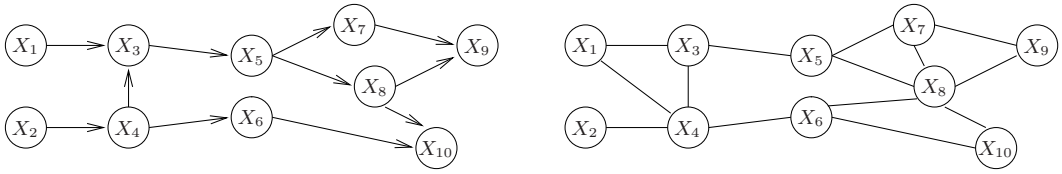
A directed graphical model, often referred to as a *Bayesian network*, constrains the distribution of a random vector X to a factored representation defined on a directed acyclic graph $\mathcal{G}$. In contrast with undirected models, there are at most n factors and each ith such factor is itself a valid conditional distribution, describing component variable X_i given its parents' variables $X_{pa(i)}$ i.e., taking $p(x_i|x_{pa(i)}) = p(x_i)$ if node i is parentless, random process X has distribution

$$p(x) = \prod_{i \in \mathcal{V}} p(x_i|x_{pa(i)}). \tag{2.11}$$

That the directed graph $\mathcal{G}$ is acyclic ensures the parent-child relationships expressed in (2.11) coincide with a well-defined partial ordering of the nodes. In turn, the random vector X is seen to realize its components sequentially in the forward partial order implied by $\mathcal{G}$.

Comparing (2.11) and (2.9), the global Markov properties implied by a directed graphical model are structurally equivalent to those implied by an undirected model with n cliques, each ith clique involving node i and its parents $pa(i)$. That is, in

[3] *Factor graphs* [58] are one way to explicitly differentiate between such specialized structures within the most general representation of (2.9), but these tools are not employed in the scope of this thesis.



(a) Directed Graphical Model (b) Equivalent Undirected Model

Figure 2.6. (a) A particular non-tree-structured directed graphical model and (b) its equivalent representation as an undirected model. The latter requires a clique for every node i and its parents $pa(i)$ in the former, which in this example adds the undirected edges $\{1,4\}$, $\{6,8\}$ and $\{7,8\}$ to the undirected counterparts of all directed edges already in (a).

general, the equivalent undirected model is defined on a graph that includes not only the undirected topology of $\mathcal{G}$, but also edges between every two nodes that share a common child; see Figure 2.6 for an example and its so-called *moral graph* [23, 60]. In turn, the local Markov properties can extend beyond the immediate neighborhood in the directed graph: in particular, the process X_i is conditionally independent of the remaining process given *both* its neighbors' variables, namely $X_{pa(i)}$ and $X_{ch(i)}$, *and* every child's other parents' variables, namely $\{X_{pa(j)\setminus i}; j \in ch(i)\}$.

The introduction of additional edges to retain an equivalent undirected model is unnecessary in the special case that $\mathcal{G}$ is a polytree. Because no two parents of the same node i share either a common ancestor or a common descendant (other than, of course, node i and its descendants), the joint distribution of all ancestors' states $X_{an(i)}$, conditioned on the local state $X_i = x_i$, can be factored across the parents' sub-polytrees i.e.,

$$p(x_{an(i)}|x_i) = \prod_{j \in pa(i)} p(x_{an(j)}, x_j|x_i), \qquad i = 1, \ldots n.$$

By marginalizing over all nodes other than node i and its parents $pa(i)$, we see that

$$p(x_{pa(i)}|x_i) = \prod_{j \in pa(i)} p(x_j|x_i), \qquad i = 1, \ldots n.$$

Substituting $p(x_i|x_{pa(i)}) \propto p(x_i) \prod_{j \in pa(i)} p(x_j|x_i)$ into (2.11), the representation specializes to exactly the pairwise form of (2.10) assuming the tree topology of $\mathcal{G}$.

■ 2.3.2 Message-Passing Algorithms on Trees

Assuming $\mathcal{G}$ is a relatively sparse graph, the representations of (2.6) along with (2.10) or (2.11) when $\mathcal{G}$ is undirected or directed, respectively, provide the means to specify

the joint distribution of a large number of (hidden and observable) random variables. Moreover, in many estimation and decision problems (including those considered in this thesis), such a specification is readily available. Typically, however, the individual factors do not readily describe the quantities of most interest for the purposes of estimation and decision-making (e.g., the posterior marginal at every node for the MPM estimation problem in Example 2.7). Indeed, for the discrete-valued hidden variables under consideration, the computation of such quantities in general graphs is known to be NP hard, scaling exponentially with the number of nodes n.

For prior probabilities $p(x)$ defined on trees, however, computation of key statistical quantities is relatively straightforward. The hidden Markov model described in Example 2.9 is the most widely studied example, for which there exist many efficient recursive algorithms, scaling linearly in n [27, 84]: there is the forward algorithm to compute the likelihood of a particular observation, the forward-backward algorithm to compute the posterior marginals, and the Viterbi algorithm to compute the posterior mode. By essentially generalizing these time series recursions to respect the partial-order implied by a tree topology, similarly efficient algorithms are available for tree-structured graphical models [79, 120]. In this subsection, we focus on so-called (sum-product) *belief propagation* algorithms [58, 79] to efficiently compute the posterior marginals at every node, addressing the MPM estimation problem described in Example 2.7.[4]

The fundamental property of a tree, which ultimately leads to the efficient recursive algorithms, is that each single node separates the graph into disjoint subtrees. More formally, for any node $i \in \mathcal{V}$ and neighbor $j \in ne(i)$, let $\mathcal{V}(j,i)$ denote the vertex set of the subtree rooted at node j looking *away* from neighbor i. Notice that $\{i\} \cup \left(\cup_{j \in ne(i)} \mathcal{V}(j,i)\right)$ comprises a disjoint union of the entire node set $\mathcal{V}$. The associated Markov properties then imply that the posterior marginal local to each node i satisfies

$$p(x_i|y) \;=\; \frac{p(y|x_i)p(x_i)}{p(y)} \;\propto\; p(x_i)p(y_i|x_i) \prod_{j \in ne(i)} p(y_{\mathcal{V}(j,i)}|x_i). \tag{2.12}$$

Thus, for the purposes of calculating $p(x_i|y)$ local to node i, the conditional likelihood $p(y_{\mathcal{V}(j,i)}|x_i)$ is a sufficient statistic of the information in the subtree associated with neighbor $j \in ne(i)$. Moreover, again applying the Markov properties implied by $\mathcal{G}$, we

[4] The key ideas are essentially the same for tree-based algorithms to obtain other statistical quantities of interest (e.g., the max-product algorithm for solving the MAP estimation problem in Example 2.6), but their details are omitted here because subsequent chapters of this thesis primarily address network-constrained analogs of the MPM estimation problem.

can relate the conditional likelihoods at neighboring nodes to one another:

$$p(y_{\mathcal{V}(j,i)}|x_i) = \frac{\sum_{x_j} p(x_i, x_j, y_{\mathcal{V}(j,i)})}{p(x_i)} = \frac{\sum_{x_j} p(x_i, x_j) p(y_{\mathcal{V}(j,i)}|x_j)}{p(x_i)}$$
$$= \sum_{x_j} p(x_j|x_i) p(y_j|x_j) \prod_{m \in ne(j) \setminus i} p(y_{\mathcal{V}(m,j)}|x_j). \quad (2.13)$$

The decompositions of (2.12) and (2.13) form the basis of a variety of algorithms for computing posterior marginals in a tree-structured graphical model. In particular, given any particular joint observation $Y = y$, we may view (2.13) as a system of nonlinear equations coupling all $2|\mathcal{E}|$ sufficient statistics, each edge (i, j) in correspondence with non-negative real-valued vectors $M_{i \to j} \in [0, \infty)^{|\mathcal{X}_j|}$ and $M_{j \to i} \in [0, \infty)^{|\mathcal{X}_i|}$. Also notice that the prior model appears in (2.13) only in terms of $p(x_j|x_i) = \frac{p(x_i,x_j)}{p(x_i)p(x_j)} p(x_j)$ for every edge in $\mathcal{G}$. In tree-structured models, these terms correspond to the canonical instance (2.7) of the more general pairwise representation (2.10). Taken as equivalent representations of the same joint distribution $p(x)$, we have

$$p(x_m|y) \propto \sum_{x_{\setminus m}} p(x, y) = \sum_{x_{\setminus m}} \prod_{i \in \mathcal{V}} p(x_i) p(y_i|x_i) \prod_{(i,j) \in \mathcal{E}} \frac{p(x_i, x_j)}{p(x_i) p(x_j)}$$
$$\propto \sum_{x_{\setminus m}} \prod_{i \in \mathcal{V}} \psi_i(x_i) p(y_i|x_i) \prod_{(i,j) \in \mathcal{E}} \psi_{i,j}(x_i, x_j).$$

It follows that satisfying the system of equations implied by (2.13) is equivalent to satisfying, up to proportionality, the system of equations implied by

$$M_{j \to i}(x_i) = \sum_{x_j} \psi_j(x_j) p(y_j|x_j) \psi_{i,j}(x_i, x_j) \prod_{m \in ne(j) \setminus i} M_{m \to j}(x_j). \quad (2.14)$$

While the statistics $M_{ne(i) \to i} = \{M_{j \to i}(x_i) \mid j \in ne(i)\}$ are, in general, no longer equal to the conditional likelihoods $\{p(y_{\mathcal{V}(j,i)}|x_i) \mid j \in ne(i)\}$, up to proportionality they remain sufficient statistics for computing the posterior marginal local to node i,

$$p(x_i|y) \propto \psi_i(x_i) p(y_i|x_i) \prod_{j \in ne(i)} M_{j \to i}(x_i). \quad (2.15)$$

Belief propagation algorithms amounts to different iterative methods for solving the system of $2|\mathcal{E}|$ equations implied by (2.14). Each sufficient statistic $M_{j \to i}(x_i)$ is viewed as a *message* that node j sends to node i, providing all the information about X_i contained in the subset of measurements $y_{\mathcal{V}(j,i)}$. One way to organize these equations is

on a node-by-node basis, where for each node i the outgoing message vectors $M_{i\to ne(i)} = \{M_{i\to j}(x_j) \mid j \in ne(i)\}$ are collectively determined by an operator, call it f_i, on the local measurement y_i and the incoming message vectors $M_{ne(i)\to i}$ i.e., (2.14) can be viewed as fixed-point equations having the form

$$M_{i\to ne(i)} = f_i\left(y_i, M_{ne(i)\to i}\right), \quad i = 1, \ldots, n. \tag{2.16}$$

Given a solution to these fixed-point equations, computing the posterior marginals $p(x_i|y)$, or the "beliefs," is straightforward via (2.15).

The parallel message schedule in belief propagation is the easiest to describe. We first initialize all messages to some arbitrary value, say $M^0_{j\to i}(x_i) = 1$ for all $i \in \mathcal{V}$ and $j \in ne(i)$. Then, we generate the sequence of messages $\{M^k_{i\to ne(i)} \mid i \in \mathcal{V}\}$ via successive so-called Jacobi iterations of (2.16), meaning all nodes update their outgoing messages in parallel based on the incoming messages of the preceding iteration i.e., iteration $k = 1, 2, \ldots$, is

$$M^k_{i\to ne(i)} := f_i\left(y_i, M^{k-1}_{ne(i)\to i}\right), \quad i = 1, \ldots, n.$$

In trees, after a number of iterations equal to the diameter of the graph (intuitively, enough iterations for data from one end of the graph to propagate to the other), the messages will converge to a unique fixed point of (2.16). One may also define the associated sequence of beliefs $\{M^k_i\}$ local to each node i, applying (2.15) after every kth message update,

$$M^k_i(x_i) \propto \psi_i(x_i)p(y_i|x_i) \prod_{j\in ne(i)} M^k_{j\to i}(x_i).$$

Each belief sequence $\{M^k_i\}$ is effectively a series of higher-fidelity approximations to the posterior marginal $p(x_i|y)$, iteratively incorporating data over an expanding neighborhood about node i in the graph $\mathcal{G}$.

From the communication perspective, associating each outgoing message to an actual transmission between two nodes, the parallel message schedule takes at least $2|\mathcal{E}|$ real-valued transmissions per iteration. It is also possible to schedule messages more efficiently, taking advantage of the partial-order implied by the tree-structured graph. In particular, by organizing the tree relative to a chosen root node (as was discussed in Example 2.10), posterior marginals can be computed via a two-pass sweep through the tree. That is, processing proceeds recursively, first from the most distant nodes to the root, and then from the root back outward (see [20, 59, 62, 79] for details). The posterior marginals can be computed after the second pass and so each individual message must only be computed once, amounting to at least $2|\mathcal{E}|$ real-valued transmissions.

The first pass is sufficient if we desire only the posterior marginal at the designated root node, amounting to total communication overhead of at least $n - 1$ real-valued transmissions.

Chapter 3

Directed Network Constraints

THIS chapter begins our deeper exploration into the connections made in Chapter 2 between Bayesian detection models and probabilistic graphical models. We reviewed that different estimation problems given an n-node graphical model (e.g., MAP estimation, likelihood calculation) can be equated with implementing the optimal centralized strategy for different cost functions given an n-sensor detection model. Efficient message-passing solutions to the former imply that communication overhead associated with the latter is at least $n-1$ real-valued messages (per online estimate), and is potentially unbounded in the absence of the ideal communication model. Even if ideal communication can be assumed for these efficient message-passing algorithms, when applied to graphical models with arbitrary graph structure, they need not necessarily lead to optimal estimates nor even converge, implying the potential for poor performance or excessive computation.

Recall from Chapter 1 that this thesis rests upon the recognition that network-constrained inference problems are characterized by two different graphs, one underlying the probabilistic model and the other underlying the communication model. The non-ideal communication models considered here take their inspiration from the efficient message-passing interpretations that exist for graphical models, while managing the twists that (i) the two graphs need not bear any relation to one another, nor even be tree-structured, (ii) every message takes values in a finite-alphabet set and (iii) the message schedule is limited to a fixed number of iterations. In following this approach, we enter the realm of approximate inference on a path that intersects with the theory of decentralized detection when the communication model imposes severe constraints (e.g., exactly one iteration with single-bit messages) in comparison to the most efficient implementations of the optimal counterpart. While in this case the online message-passing algorithms, or equivalently decentralized decision strategies, are then efficient and convergent by constraint, the key challenges arise in tractably solving the associated

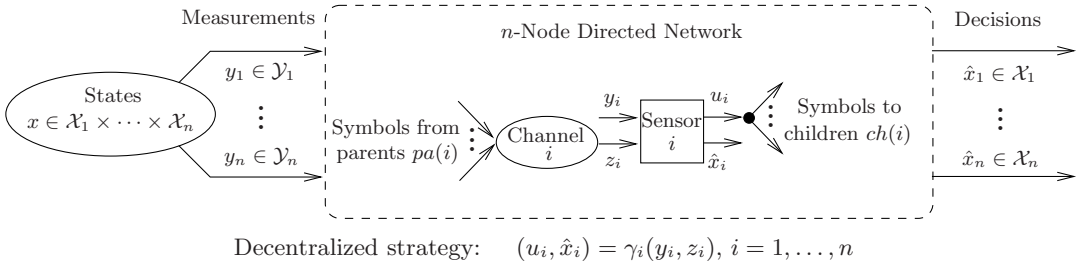


Figure 3.1. The n-sensor detection model described in Chapter 2, but assuming a decentralized decision strategy subject to network constraints defined on an n-th order directed acyclic graph, each edge representing a unidirectional finite-rate (and perhaps unreliable) communication link between two spatially-distributed nodes. The online message schedule is constrained to just a single forward sweep through the network, each node successively receiving information from its parents (if any), transmitting information to its children (if any), and forming a local state estimate (if in the gateway).

offline design problems, optimizing over the set of *feasible* strategies such that the loss from optimal centralized performance is minimized.

■ 3.1 Chapter Overview

This chapter focuses on the non-ideal communication model in which the online message schedule is constrained to exactly one forward sweep on a given directed acyclic graph. As illustrated in Figure 3.1, each node successively receives information from its parents, may transmit at most one discrete symbol to its children, and then may form its own local state estimate. In the special case that the probabilistic graphical model and directed network topology have identical tree structure, this online message schedule can be viewed as the quantized analog to the forward-sweep algorithm for calculating exact likelihoods at all childless nodes as discussed in Chapter 2. The variational formulation and team-theoretic analysis presented in this chapter generalizes a number of previous studies in the decentralized detection literature [11, 106, 109, 110], including the consideration of an arbitrary directed network topology [26, 81, 96, 97, 98], a vector state process along with a distributed decision objective [80, 81], as well as selective or unreliable online communication [16, 17, 74, 83, 88].

Section 3.2 augments the n-sensor detection formulation to account for the directed network constraints in the generality implied by Figure 3.1. Existing team theory establishes when necessary optimality conditions reduce to a convergent iterative algorithm to be executed offline. While the resulting online strategy admits an efficient distributed implementation by design, without introducing additional structure the associated of-

fline algorithm has exponential complexity in the number of nodes, and its distributed implementation assumes every node will iteratively broadcast to all other nodes in the network.

In Section 3.3, we identify a class of models for which the convergent offline algorithm itself admits an efficient message-passing interpretation on the given network topology. In each offline iteration, every node adjusts its local rule (for subsequent online processing) based on incoming messages from its neighbors and, in turn, sends adjusted outgoing messages to its neighbors. The messages received by each node from its parents define, in the context of its local objectives, a "likelihood function" for the symbols it may receive online (e.g., "what does the information from my neighbors mean to me") while the messages from its children define, in the context of all other nodes' objectives, a "cost-to-go function" for the symbols it may transmit online (e.g., "what does the information from me mean to my children and their descendents"). Each node need only be initialized with local statistics and iterative per-node computation becomes invariant to n (but still scales exponentially with the number of neighbors, so the algorithm is best-suited for sparsely-connected networks).

The end result of this offline message-passing process can be thought of as a distributed *fusion protocol*, in which the nodes of the network have collectively determined their individual rules for transmitting information to their children and interpreting information transmitted by their parents. As we will illustrate, this protocol takes into account explicitly the limits on available communication resources, in effect using the absence of communication as another noisy signal from one node to another, which we show can be of value even when communication channels are unreliable or communication costs are negligible. In addition, the prospect of a computationally-efficient algorithm to optimize large-scale decentralized detection networks is complementary to other recent work, which focuses on asymptotic analyses [1, 15, 75, 101, 102, 105, 122], typically under assumptions regarding network regularity or sensor homogeneity. The message-passing algorithm we propose here may offer a tractable design alternative for applications in which such assumptions cannot be made, and especially if network connectivity is also sparse and the detection objective is itself spatially-distributed.

Section 3.4 describes a number of experiments with a simulated network of binary detectors, applying the offline message-passing algorithm to optimize the achievable tradeoff between global detection performance and network-wide online communication. The results illustrate that, considering the severity of the online communication constraints, relatively dramatic improvements over myopic decentralized performance

are possible. Our empirical analysis also exposes a design tradeoff between constraining in-network processing to conserve algorithmic resources (per online measurement) but then having to consume resources (per offline organization) to maintain detection performance.

Section 3.5 closes this chapter by summarizing these results in preparation for their extension to the more elaborate online message schedules considered in subsequent chapters. Chapter 4 focuses on network constraints defined on an undirected graph, showing that, under certain assumptions, the problem is structurally equivalent to that addressed in this chapter. In Chapter 5, we formulate the possibility of multiple online communication stages: its team solution turns out to be intractable even under best-case assumptions, but the offline message-passing interpretations developed for the single-stage communication architectures form an integral part of the approximations we propose to tackle these difficult problems.

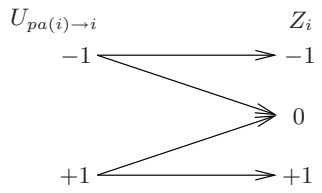
■ 3.2 Decentralized Detection Networks

This section reviews the theory of decentralized Bayesian detection [106, 109, 110] in the generality implied by Figure 3.1. Our main model builds upon the n-sensor detection model discussed in Chapter 2. As before, we first assume (i) the hidden state x and observable measurement y take their values in, respectively, a discrete product space $\mathcal{X} = \mathcal{X}_1 \times \cdots \times X_n$ and Euclidean product space $\mathcal{Y} = \mathcal{Y}_1 \times \cdots \times \mathcal{Y}_n$. We assume a given distribution $p(x, y)$ jointly describes the hidden state process X and noisy measurement process Y. Note that an m-ary hypothesis test can be viewed as a special of this model, corresponding to $|\mathcal{X}_i| = m$ for every i and prior probabilities $p(x)$ such that $\mathbf{P}[X_1 = X_2 = \cdots = X_n] = 1$. Different from before, we assume the global estimate $\hat{x} \in \mathcal{X}$ is generated sequentially in the forward partial order of a given n-node directed acyclic graph $\mathcal{F} = (\mathcal{V}, \mathcal{D})$, each edge (i, j) in $\mathcal{F}$ indicating a (perhaps unreliable) low-rate communication link from node i to node j. As illustrated in Figure 3.1, each node i, observing only the component measurement y_i and the symbol(s) z_i received on incoming links with all parents $pa(i) = \{j \in \mathcal{V} \mid (j, i) \in \mathcal{D}\}$ (if any), is to decide upon both its component estimate $\hat{x}_i$ and the symbol(s) u_i transmitted on outgoing links with all children $ch(i) = \{j \in \mathcal{V} \mid (i, j) \in \mathcal{D}\}$ (if any). We now proceed to more carefully formulate such online processing constraints, translating them to explicit restrictions on the set of decision strategies over which the Bayes risk function is minimized.

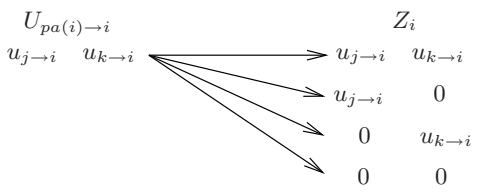
■ 3.2.1 Network-Constrained Online Processing Model

Suppose each edge (i,j) in $\mathcal{F}$ is assigned an integer $d_{i\to j} \geq 2$ denoting the size of the symbol set supported by this link (i.e., the link rate is $\log_2 d_{i\to j}$ bits per measurement). The symbol(s) u_i transmitted by node i can thus take *at most* $\prod_{j\in ch(i)} d_{i\to j}$ distinct values. For example, a scheme in which node i may transmit a different symbol to each child is modeled by a finite set $\mathcal{U}_i$ with cardinality equal to $\prod_{j\in ch(i)} d_{i\to j}$, while a scheme in which node i transmits the same symbol to every child corresponds to $|\mathcal{U}_i| = \min_{j\in ch(i)} d_{i\to j}$. In any case, the focus is on models that require each node to somehow compress its local data into a relatively small number of logical outgoing symbols (e.g., one symbol per outgoing link). We similarly assume the symbol(s) z_i received by node i take their values in a given discrete set $\mathcal{Z}_i$. The cardinality of $\mathcal{Z}_i$ will certainly reflect the joint cardinality $|\mathcal{U}_{pa(i)}| = \prod_{j\in pa(i)} |\mathcal{U}_j|$ of its parents' transmissions, but the exact relation is determined by the given multipoint-to-point channel into each node i. In any case, each such channel is modeled by a conditional distribution $p(z_i|x,y,u_{pa(i)})$, describing the information Z_i received by node i based on its parents' transmitted symbols $u_{pa(i)} = \{u_j \in \mathcal{U}_j \mid j \in pa(i)\}$.[1]

Example 3.1 (Peer-to-Peer Binary Communication with Erasures). Associate each edge (i,j) in directed graph $\mathcal{F}$ with a unit-rate communication link, meaning $d_{i\to j} = 2$. If $u_{i\to j} \in \{-1,+1\}$ denotes the actual symbol transmitted by node i to its child $j \in ch(i)$, then the collective communication decision u_i takes its values in $\mathcal{U}_i = \{-1,+1\}^{|ch(i)|}$. The collection of all symbols transmitted to a particular node j is denoted by $u_{pa(j)\to j} = \{u_{i\to j}; i \in pa(j)\}$. On the receiving end, let $z_{j\to i} \in \{-1,0,+1\}$ denote the actual symbol received by node i from its parent $j \in pa(i)$, where the value "0" indicates an erasure and otherwise $z_{j\to i} = u_{j\to i}$. It follows that the symbol z_i received by node i takes values in $\mathcal{Z}_i = \{-1,0,+1\}^{|pa(i)|}$.



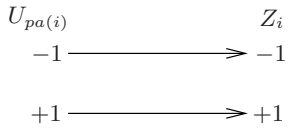
Given Node i has One Parent



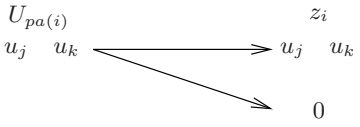
Given Node i has Two Parents

[1] Here, we have also allowed the channel model to depend on the processes (X,Y) of the environment external to the network. Whether such generality is warranted will, of course, depend on the application (e.g., the sensor seeks to detect the presence of a malicious jammer), and later sections will indeed sacrifice some generality in the interest of scalable representations and tractable algorithms.

Example 3.2 (Broadcast Binary Communication with Interference). As in Example 3.1, let $d_{i\to j} = 2$ for each edge (i, j) in $\mathcal{F}$. However, now assume each node i always transmits the same binary-valued symbol to all of its children, meaning $\mathcal{U}_i = \{-1, +1\}$. On the receiving end there are two possibilities: either $z_i = u_{pa(i)}$ or, when there are two or more parents, none of the incoming symbols are received due to inter-symbol interference. Denoting the latter event by $z_i = 0$, it follows that $\mathcal{Z}_i = \{-1, +1\}^{|pa(i)|} \times \{0\}$.



Given Node i has One Parent



Given Node i has Two Parents

Altogether, the collections of transmitted symbols u and received symbols z thus take their values in discrete product spaces $\mathcal{U} = \mathcal{U}_1 \times \cdots \times \mathcal{U}_n$ and $\mathcal{Z} = \mathcal{Z}_1 \times \cdots \times \mathcal{Z}_n$, respectively. By constraint, the global decision process $\hat{X}$ is generated in a component-wise fashion in the forward partial order of network topology $\mathcal{F}$, each node i individually generating both U_i and $\hat{X}_i$ upon observing both Y_i and Z_i. It follows that any particular *strategy* $\gamma : \mathcal{Y} \times \mathcal{Z} \to \mathcal{U} \times \mathcal{X}$ induces a global decision process $(U, \hat{X}) = \gamma(Y, Z)$. Denote by $\bar{\Gamma}$ the set of all such strategies and by $\Gamma \subset \bar{\Gamma}$ the *admissible*, or feasible, subset of these strategies given the network topology $\mathcal{F}$; specifically, denoting by Γ_i the set of all rules $\gamma_i : \mathcal{Y}_i \times \mathcal{Z}_i \to \mathcal{U}_i \times \mathcal{X}_i$ local to node i, we let $\Gamma = \Gamma_1 \times \cdots \times \Gamma_n$.

■ 3.2.2 Bayesian Formulation with Costly Communication

The Bayesian criterion is essentially the same as in the centralized detection problem, but also accounting for the communication-related decision process U. We assign to every possible realization of the joint process $(U, \hat{X}, X)$ a cost of the form

$$c(u, \hat{x}, x) = c(\hat{x}, x) + \lambda c(u, x),$$

where non-negative constant λ specifies the unit conversion between detection costs $c(\hat{x}, x)$ and communication costs $c(u, x)$. In turn, the Bayes risk function is given by

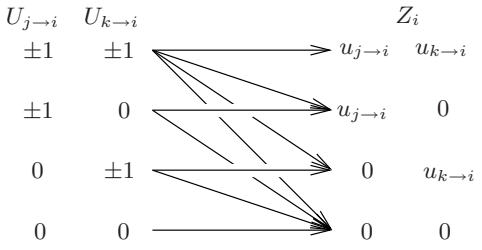
$$J(\gamma) = \mathbf{E}\left[c(U, \hat{X}, X)\right] = \mathbf{E}\left[\mathbf{E}\left[c(\gamma(Y, Z), X)\middle| Y, Z\right]\right] \tag{3.1}$$

and the decentralized design problem is to find the strategy $\gamma^* \in \Gamma \subset \bar{\Gamma}$ such that

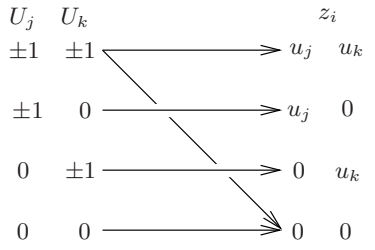
$$\begin{aligned} J(\gamma^*) &= J_d(\gamma^*) + \lambda J_c(\gamma^*) \\ &= \min_{\gamma \in \bar{\Gamma}} J_d(\gamma) + \lambda J_c(\gamma) \text{ subject to } \gamma \in \Gamma, \end{aligned} \tag{3.2}$$

where functions $J_d : \bar{\Gamma} \to \mathbb{R}$ and $J_c : \bar{\Gamma} \to \mathbb{R}$ quantify the *detection penalty* and *communication penalty*, respectively. Viewing (3.2) as a multi-objective criterion parameterized by λ, the achievable design tradeoff is then captured by the *pareto-optimal* planar curve $\{(J_c(\gamma^*), J_d(\gamma^*)) \, ; \lambda \geq 0\}$.

Example 3.3 (Selective Binary Communication Schemes). As in Example 3.1 and Example 3.2, let $d_{i\to j} = 2$ for each edge (i, j) in $\mathcal{F}$. A selective communication scheme refers to each node having the option to suppress transmission on, or remain silent, on one or more of its outgoing links. We denote this option to remain silent by the symbol "0", and we assume it is always both cost-free and reliably received. In Example 3.1, for example, this implies any communicating node i selects from an augmented decision space of $\mathcal{U}_i = \{-1, 0, +1\}^{|pa(i)|}$. Meanwhile, upon receiving $z_{i\to j} = 0$, any child $j \in ch(i)$ is then uncertain as to whether node i elected silence or link (i, j) experienced an erasure; on the other hand, if $z_{i\to j} \neq 0$, then child j knows neither selective silence nor an erasure has occurred. In Example 3.1, we let $\mathcal{U}_i = \{-1, 0, +1\}$ for node i, while on the receiving end the effects of interference occur only among the subset of actively transmitting parents.



Selective Communication in Example 3.1



Selective Communication in Example 3.2

The formulation in (3.2) specializes to the centralized design problem when online communication is both unconstrained and unpenalized i.e., Γ is the set of all functions $\gamma : \mathcal{Y} \to \mathcal{X}$ and $\lambda = 0$. In general, however, the function space Γ excludes the optimal centralized strategy $\bar{\gamma}$ in (2.2), but always includes the myopic decentralized strategy $\bar{\delta}$ in (2.4). The non-ideal communication model also manifests itself as a factored representation within the distribution underlying (3.1). By construction, fixing a rule $\gamma_i \in \Gamma_i$ is equivalent to specifying the distribution

$$p(u_i, \hat{x}_i | y_i, z_i; \gamma_i) = \begin{cases} 1 & , \quad \text{if } (u_i, \hat{x}_i) = \gamma_i(y_i, z_i) \\ 0 & , \quad \text{otherwise} \end{cases} .$$

It follows that fixing a strategy $\gamma \in \Gamma \subset \bar{\Gamma}$ specifies the distribution

$$p(u, z, \hat{x} | x, y; \gamma) = \prod_{i=1}^{n} p(z_i | x, y, u_{pa(i)}) p(u_i, \hat{x}_i | y_i, z_i; \gamma_i), \tag{3.3}$$

reflecting the causal processing implied by the directed network topology $\mathcal{F}$. In turn, the distribution that determines the global penalty function $J(\gamma)$ in (3.1) becomes

$$p(u, \hat{x}, x; \gamma) = \int_{y \in \mathcal{Y}} p(x, y) \prod_{i=1}^{n} p(u_i, \hat{x}_i | x, y, u_{pa(i)}; \gamma_i)\, dy, \tag{3.4}$$

where the summation over $\mathcal{Z}$ is taken inside the product i.e., for each node i, we have

$$p(u_i, \hat{x}_i | x, y, u_{pa(i)}; \gamma_i) = \sum_{z_i \in \mathcal{Z}_i} p(z_i | x, y, u_{pa(i)}) p(u_i, \hat{x}_i | y_i, z_i; \gamma_i).$$

Note that the integration over $\mathcal{Y}$ cannot be decomposed in the absence of additional model assumptions, a possibility we explore subsequently.

■ 3.2.3 Team-Theoretic Solution

In general, it is not known whether the strategy γ^* in (3.2) lies in a finitely-parameterized subspace of Γ. The team-theoretic approximation used here is to satisfy a set of *person-by-person* optimality conditions, each based on a simple observation: if a decentralized strategy $\gamma^* = (\gamma_1^*, \ldots, \gamma_n^*)$ is optimal over Γ, then for each i and assuming rules $\gamma_{\setminus i}^* = \{\gamma_j^* \in \Gamma_j \mid j \neq i\}$ are fixed, the rule γ_i^* is optimal over Γ_i i.e., for each i,

$$\gamma_i^* \quad = \quad \arg\min_{\gamma_i \in \Gamma_i} J_d(\gamma_{\setminus i}^*, \gamma_i) + \lambda J_c(\gamma_{\setminus i}^*, \gamma_i). \tag{3.5}$$

Simultaneously satisfying (3.5) for all i is (by definition) a necessary optimality condition, but it is not sufficient because, in general, it does not preclude a decrease in J via joint minimization over multiple nodes simultaneously. Under certain model assumptions, finding a solution to the n coupled optimization problems in (3.5) reduces analytically to finding a fixed-point of a particular system of nonlinear equations.

In this and subsequent sections we introduce a sequence of further model assumptions, each of which introduces additional local structure to our problem which we exploit in constructing our efficient iterative offline algorithm. We do this in stages to help elucidate the value and impact of each of these successive assumptions.

Assumption 3.1 (Conditional Independence)**.** *Conditioned on the state process X, the measurement Y_i and received symbol Z_i local to node i are mutually independent as well as independent of all other information observed in the network, namely the measurements $Y_{\setminus i}$ and symbols $Z_{\setminus i}$ received by all other nodes i.e., for every i,*

$$p(y_i, z_i | x, y_{\setminus i}, z_{\setminus i}, u_{\setminus i}) = p(y_i | x) p(z_i | x, u_{pa(i)}). \tag{3.6}$$

For example, Assumption 3.1 is satisfied if each measurement Y_i is a function of X corrupted by noise, each received symbol Z_i is a function of X (and transmitted symbols $U_{pa(i)}$) corrupted by noise, and all of these noise processes are mutually independent.

Lemma 3.1 (Factored Representation)**.** *Let Assumption 3.1 hold. For every strategy $\gamma \in \Gamma$, the distribution in (3.4) specializes to*

$$p(u, \hat{x}, x; \gamma) = p(x) \prod_{i=1}^{n} p(u_i, \hat{x}_i | x, u_{pa(i)}; \gamma_i),$$

where for every i,

$$p(u_i, \hat{x}_i | x, u_{pa(i)}; \gamma_i) = \sum_{z_i \in \mathcal{Z}_i} p(z_i | x, u_{pa(i)}) \int_{y_i \in \mathcal{Y}_i} p(y_i | x) p(u_i, \hat{x}_i | y_i, z_i; \gamma_i) \, dy_i. \tag{3.7}$$

Proof. Substituting (3.6) into (3.3) and (3.4) results in

$$p(u, \hat{x} | x; \gamma) = \sum_{z \in \mathcal{Z}} \int_{y \in \mathcal{Y}} \prod_{i=1}^{n} p(y_i | x) p(z_i | x, u_{pa(i)}) p(u_i, \hat{x}_i | y_i, z_i; \gamma_i) \, dy.$$

Because only the ith factor in the integrand involves variables (y_i, z_i), global marginalization over (Y, Z) simplifies to n local marginalizations, each over (Y_i, Z_i). □

Proposition 3.1 (Person-by-Person Optimality)**.** *Let Assumption 3.1 hold. The ith component optimization in (3.5) reduces to*

$$\gamma_i^*(Y_i, Z_i) = \arg \min_{(u_i, \hat{x}_i) \in \mathcal{U}_i \times \mathcal{X}_i} \sum_{x \in \mathcal{X}} \theta_i^*(u_i, \hat{x}_i, x; Z_i) p(Y_i | x) \tag{3.8}$$

where, for each $z_i \in \mathcal{Z}_i$ such that $p(Y_i, z_i; \gamma_{\setminus i}^*) > 0$, the parameter values $\theta_i^*(z_i) \in \mathbb{R}^{|\mathcal{U}_i| \times |\mathcal{X}_i| \times |\mathcal{X}|}$ are given by

$$\theta_i^*(u_i, \hat{x}_i, x; z_i) = p(x) \sum_{u_{\setminus i} \in \mathcal{U}_{\setminus i}} p(z_i | x, u_{pa(i)}) \sum_{\hat{x}_{\setminus i} \in \mathcal{X}_{\setminus i}} c(u, \hat{x}, x) \prod_{j \neq i} p(u_j, \hat{x}_j | x, u_{pa(j)}; \gamma_j^*). \tag{3.9}$$

Proof. The proof follows the same key steps by which (2.2) is derived in the centralized case, but accounting for a composite measurement (Y_i, Z_i) and a cost function that also depends on non-local decision variables $(U_{\setminus i}, \hat{X}_{\setminus i})$. Assumption 3.1 is essential for the parameter values θ_i^* to be independent of the local measurement Y_i. See Appendix A.1. □

It is instructive to note the similarity between a local rule γ_i^* in Proposition 3.1 and the centralized strategy in (2.2). Both process an $|\mathcal{X}|$-dimensional sufficient statistic of the available measurement with optimal parameter values to be computed offline. In rule γ_i^*, however, the offline computation is more than simple multiplication of probabilities $p(x)$ and costs $c(u, \hat{x}, x)$: parameter values $\theta_i^* \in \mathbb{R}^{|\mathcal{U}_i|\times|\mathcal{X}_i|\times|\mathcal{X}|\times|\mathcal{Z}_i|}$ in (3.9) now involve conditional expectations, taken over distributions that depend on the fixed rules $\gamma_{\setminus i}^*$ of all other nodes $j \neq i$. Each such fixed rule γ_j^* is similarly of the form in Proposition 3.1, where fixing parameter values θ_j^* specifies $p(u_j, \hat{x}_j|x, u_{pa(j)}; \theta_j^*)$ local to node j through (3.7) and (3.8). Each ith minimization in (3.5) is thereby equivalent to minimizing

$$J(\gamma_{\setminus i}^*, \gamma_i) = \sum_{x\in\mathcal{X}} p(x) \sum_{u\in\mathcal{U}} \sum_{\hat{x}\in\mathcal{X}} c(u, \hat{x}, x) p(u, \hat{x}|x; \theta_{\setminus i}^*, \theta_i)$$

over the parameterized space of distributions defined by

$$p(u, \hat{x}|x; \theta_{\setminus i}^*, \theta_i) = p(u_i, \hat{x}_i|x, u_{pa(i)}, \theta_i) \prod_{j\neq i} p(u_j, \hat{x}_j|x, u_{pa(j)}; \theta_j^*).$$

It follows that the simultaneous satisfaction of (3.5) at all nodes corresponds to solving for $\theta^* = (\theta_1^*, \ldots, \theta_n^*)$ in a system of nonlinear equations expressed by (3.7)–(3.9). Specifically, if we let $f_i(\theta_{\setminus i}^*)$ denote the right-hand-side of (3.9), then offline computation of a person-by-person optimal strategy reduces to solving the fixed-point equations

$$\theta_i = f_i(\theta_{\setminus i}), \quad i = 1, \ldots, n. \tag{3.10}$$

Corollary 3.1 (Offline Iterative Algorithm)**.** *Initialize parameters $\theta^0 = (\theta_1^0, \ldots, \theta_n^0)$ and generate the sequence $\{\theta^k\}$ by iterating (3.10) in any component-by-component order e.g., iteration $k = 1, 2, \ldots$ is*

$$\theta_i^k := f_i(\theta_1^k, \ldots, \theta_{i-1}^k, \theta_{i+1}^{k-1}, \ldots, \theta_n^{k-1}), \quad i = 1, \ldots, n.$$

If Assumption 3.1 holds, then the associated sequence $\{J(\gamma^k)\}$ is non-increasing and converges.

Proof. By virtue of Proposition 3.1, each operator f_i is the solution to the minimization of J over the ith coordinate function space Γ_i. Any component-wise iteration of f is thus equivalent to a coordinate-descent iteration of J, implying $J(\gamma^k) \leq J(\gamma^{k-1})$ for every k [6]. Because the real-valued, non-increasing sequence $\{J(\gamma^k)\}$ is also bounded below, it has a limit point. □

In the absence of additional technical conditions (e.g., J is convex, f is contracting [6]), it is *not* known whether the sequence $\{J(\gamma^k)\}$ converges to the optimal performance $J(\gamma^*)$, whether the achieved performance is invariant to the choice of initial parameters θ^0, nor whether the associated sequence $\{\theta^k\}$ converges. Indeed, the possibility of a poorly performing person-by-person-optimal strategy is known to exist (see [48] and [21] for such crafted special cases). These theoretical limitations are inherent to nonlinear minimization problems, in general, where second-order optimality conditions can be "locally" satisfied at many points, but only one of them may achieve the "global" minimum. Nonetheless, the iterative algorithm is often reported to yield reasonable decision strategies, which has also been our experience (in experiments to be described) providing the iterative algorithm is initialized with some care.

Also note that Corollary 3.1 assume every node i can *exactly* compute the local marginalization of (3.7). Some measurement models of practical interest lead to numerical or Monte-Carlo approximation of these marginalizations at each iteration k, and the extent to which the resulting errors may affect convergence is also not known. This issue is beyond the scope of this thesis and, as such, all of our experiments will involve sensor models in which such complications do not arise (e.g., the models in Example 2.5 and Example 2.8).

■ 3.3 Efficient Message-Passing Interpretations

Online measurement processing implied by Proposition 3.1 is, by design, well-suited for distributed implementation. However, a number of practical difficulties remain:

- convergent offline optimization requires global knowledge of probabilities $p(x)$, costs $c(u, \hat{x}, x)$ and statistics $\{p(u_i, \hat{x}_i | x, u_{pa(i)}; \theta_i^k)\}$ in every iteration k;
- total (offline and online) memory/computation requirements scale exponentially with the number of nodes n.

In this section, we establish conditions so that convergent offline optimization can be executed in a recursive fashion: each node i starts with local probabilities $p(x_{pa(i)}, x_i)$ and local costs $c(u_i, \hat{x}_i, x_i)$, then in each iteration computes and exchanges rule-dependent statistics, or *messages*, with only its neighbors $pa(i) \cup ch(i)$. We will interpret this message-passing algorithm as an instance of Corollary 3.1 under some additional model assumptions. Thus, when these additional assumptions hold, it inherits the same theoretical convergence properties. Moreover, we will see that total memory/computation requirements scale only linearly with n.

The decentralized detection formulation discussed in Section 3.2 belongs to the class of (static, discrete) team decision problems, which have been studied for many decades in both the economics and engineering literature. The message-passing algorithm described in the following subsections is primarily built upon the computational theory discussed in [80, 81, 96, 97, 98, 106], albeit each of these considers only certain special cases of Figure 3.1. For example, both [98] and [106] develop Proposition 3.1 and Corollary 3.1 assuming a global binary hypothesis test and the parallel network topology (i.e., a set of mutually-disconnected peripheral nodes reliably connected to a common fusion node). Both [106] and [97] extend the analysis to a singly-rooted tree topology, assuming the objective is for just the root node to make the final binary-valued decision with minimum error probability. The extension to problems in which the discrete state process X is itself spatially-distributed, in the sense that a different state variable X_i is associated with each node i, has been studied in [80, 81].

One contribution of the development in this chapter is the generality with which the results apply. For example, the efficient algorithm proposed in [97] is a special case of our message-passing algorithm; yet our derivation need not assume *from the start* that all nodes must employ local likelihood-ratio tests, nor that the penalty function J is differentiable with respect to the threshold parameters. Our general development also incorporates the possibility of a selective (or censored) transmission scheme and unreliable communication channels, aspects also considered for the parallel network topology with a global fusion node in [88] and [16], respectively. Our main contribution, however, stems from our emphasis not just on preserving algorithm correctness as we make these generalizations, but also on preserving algorithmic efficiency. As will be discussed, an important new insight provided by our analysis is the extent to which the graphical structure underlying the distributed state process may deviate from the communication network topology without sacrificing either algorithm correctness or efficiency. Moreover, the local recursive structure of the message-passing equations can be applied to network topologies beyond those for which it is originally derived, providing a new approximation paradigm for large irregular networks of heterogeneous sensors in which the general algorithm of Corollary 3.1 is intractable and conclusions based on asymptotic analyses are not readily available.

■ 3.3.1 Online Measurement Processing

We first introduce an assumption that removes the exponential dependence on the number of nodes n of the online computation i.e., the actual operation of the optimized

strategy as data are received and communication and decision-making takes place. This exponential dependence is due to the appearance of the global state vector X in (3.8). The following assumption reduces this to a dependence only on the local state component of each node.

Assumption 3.2 (Measurement/Channel Locality)**.** *In addition to the conditions of Assumption 3.1, the measurement and channel models local to node i do not directly depend on any of the non-local state processes $X_{\setminus i}$ i.e., for every i,*

$$p(y_i, z_i|x, y_{\setminus i}, z_{\setminus i}, u_{\setminus i}) = p(y_i|x_i)p(z_i|x_i, u_{pa(i)}). \tag{3.11}$$

Corollary 3.2 (Online Efficiency)**.** *If Assumption 3.2 holds, then (3.8) and (3.9) in Proposition 3.1 specialize to*

$$\gamma_i^*(Y_i, Z_i) = \arg \min_{(u_i,\hat{x}_i)\in\mathcal{U}_i\times\mathcal{X}_i} \sum_{x_i\in\mathcal{X}_i} \phi_i^*(u_i, \hat{x}_i, x_i; Z_i)p(Y_i|x_i) \tag{3.12}$$

and

$$\phi_i^*(u_i, \hat{x}_i, x_i; z_i) = \sum_{x_{\setminus i}} p(x) \sum_{u_{\setminus i}} p(z_i|x_i, u_{pa(i)}) \sum_{\hat{x}_{\setminus i}} c(u, \hat{x}, x) \prod_{j\neq i} p(u_j, \hat{x}_j|x_j, u_{pa(j)}; \gamma_j^*), \tag{3.13}$$

respectively.

Proof. Recognizing (3.11) to be the special case of (3.6) with $p(y_i|x) = p(y_i|x_i)$ and $p(z_i|x, u_{pa(i)}) = p(z_i|x_i, u_{pa(i)})$ for every i, (3.7) in Lemma 3.1 similarly specializes to

$$p(u_i, \hat{x}_i|x_i, u_{pa(i)}; \gamma_i) = \sum_{z_i\in\mathcal{Z}_i} p(z_i|x_i, u_{pa(i)}) \int_{y_i\in\mathcal{Y}_i} p(y_i|x_i)p(u_i, \hat{x}_i|y_i, z_i; \gamma_i)\, dy_i \tag{3.14}$$

for every i. We then apply Proposition 3.1 with

$$\phi_i^*(u_i, \hat{x}_i, x_i; z_i) = \sum_{x_{\setminus i}\in\mathcal{X}_{\setminus i}} \theta_i^*(u_i, \hat{x}_i, x; z_i).$$

□

It is instructive to note the similarity between γ_i^* in Corollary 3.2 and the local myopic rule $\bar{\delta}_i$ in (2.4). Online computation is nearly identical, but with γ_i^* using parameters that reflect the composite decision space $\mathcal{U}_i\times\mathcal{X}_i$ and depend explicitly on the received information $Z_i = z_i$. This similarity is also apparent in the offline computation implied by (3.14) for fixed parameters ϕ_i^* in (3.12), which per value $z_i \in \mathcal{Z}_i$ involves the same local marginalization over Y_i highlighted for fixed parameters $\bar{\phi}_i$ in (2.5).

■ 3.3.2 Offline Strategy Optimization

Efficiency in the offline iterative algorithm—i.e., in the algorithm for computing the optimized decision rules at each node—requires not only the locality of the measurements and channels as in Assumption 3.2 but a bit more, namely that the overall cost function decomposes into a sum of per-node local costs, and that network topology is a polytree.

Assumption 3.3 (Cost Locality)**.** *The Bayesian cost function is additive across the nodes of the network i.e.,*

$$c(u, \hat{x}, x) = \sum_{i=1}^{n} c(u_i, \hat{x}_i, x_i). \tag{3.15}$$

Assumption 3.4 (Polytree Topology)**.** *Directed graph $\mathcal{F}$ is a polytree i.e., there is at most one (directed) path between any pair of nodes.*

Proposition 3.2 (Offline Efficiency)**.** *If Assumptions 3.2–3.4 hold, then (3.12) applies with (3.13) specialized to the proportionality*

$$\phi_i^*(u_i, \hat{x}_i, x_i; z_i) \propto p(x_i) P_i^*(z_i|x_i) \left[c(u_i, \hat{x}_i, x_i) + C_i^*(u_i, x_i) \right], \tag{3.16}$$

where (i) the likelihood function $P_i(z_i|x_i)$ *for received information Z_i is determined by the forward recursion*

$$P_i^*(z_i|x_i) = \begin{cases} 1 & , \quad pa(i) \text{ empty} \\ \displaystyle\sum_{x_{pa(i)}} \sum_{u_{pa(i)}} p(x_{pa(i)}|x_i) p(z_i|x_i, u_{pa(i)}) \prod_{j \in pa(i)} P_{j\to i}^*(u_j|x_j) & , \quad \text{otherwise} \end{cases} \tag{3.17}$$

with the forward message *from each parent $j \in pa(i)$ given by*

$$P_{j\to i}^*(u_j|x_j) = \sum_{z_j} P_j^*(z_j|x_j) \sum_{\hat{x}_j} p(u_j, \hat{x}_j|x_j, z_j; \gamma_j^*), \tag{3.18}$$

and (ii) the cost-to-go function $C_i(u_i, x_i)$ *for transmitted information U_i is determined by the backward recursion*

$$C_i^*(u_i, x_i) = \begin{cases} 0 & , \quad ch(i) \text{ empty} \\ \displaystyle\sum_{j \in ch(i)} C_{j\to i}^*(u_i, x_i) & , \quad \text{otherwise} \end{cases} \tag{3.19}$$

with the backward message *from each child* $j \in ch(i)$ *given by*

$$C^*_{j\to i}(u_i, x_i) = \sum_{x_j} \sum_{u_j} \sum_{\hat{x}_j} \left[c(u_j, \hat{x}_j, x_j) + C^*_j(u_j, x_j)\right] Q^*_{j\to i}(u_j, \hat{x}_j, x_j | u_i, x_i) \qquad (3.20)$$

with

$$\begin{aligned} Q^*_{j\to i}(u_j, \hat{x}_j, x_j | u_i, x_i) &= \sum_{x_{pa(j)\setminus i}} p(x_{pa(j)}, x_j | x_i) R^*_{j\to i}(u_j, \hat{x}_j | x_j, x_{pa(j)\setminus i}), \\ R^*_{j\to i}(u_j, \hat{x}_j | x_j, x_{pa(j)\setminus i}) &= \sum_{u_{pa(j)\setminus i}} p(u_j, \hat{x}_j | x_j, u_{pa(j)}; \gamma^*_j) \prod_{m\in pa(j)\setminus i} P^*_{m\to j}(u_m | x_m). \end{aligned}$$

Proof. We provide only the sketch here; see Appendix A.2 for details. By virtue of Assumption 3.2, the global likelihood function for received information Z_i is independent of the rules and states local to nodes other than i and its *ancestors* (i.e., the parents $pa(i)$, each such parent's parents, and so on). By virtue of Assumption 3.3, the global penalty function itself takes an additive form over all nodes, where terms local to nodes other than i and its *descendants* (i.e., the children $ch(i)$, each such child's children, and so on) cannot be influenced by local decision $(u_i, \hat{x}_i)$ and, hence, have no bearing on the optimization of rule γ_i. By virtue of Assumption 3.4, the information observed and generated by all ancestors is independent (conditioned on X while optimizing γ_i) of the information *to be* observed and generated by all descendents. This conditional independence between the "upstream" likelihood statistics and the "downstream" expected costs specializes the parameter values ϕ^*_i of Corollary 3.2 to the particular form of (3.16). Assumption 3.4 also guarantees no two parents have a common ancestor, implying that upstream likelihoods decompose multiplicatively across parent nodes, and similarly no two children have a common descendant, implying that downstream costs decompose additively across child nodes. Altogether, Assumptions 3.2–3.4 and their respective structural implications yield the recursive formulas expressed by (3.17)–(3.20). □

Proposition 3.2 has a number of important implications. The first is that parameters ϕ^*_i at node i are now completely determined by the incoming messages from its neighbors $pa(i) \cup ch(i)$. Specifically, we see in (3.16) that the global meaning of received information Z_i manifests itself as a Bayesian correction to the myopic prior $p(x_i)$, while the global meaning of transmitted information U_i manifests itself as an additive correction to the myopic cost $c(u_i, \hat{x}_i, x_i)$. The former correction requires the likelihood function P^*_i expressed by (3.17), uniquely determined from the incoming forward messages $P^*_{pa(i)\to i} = \{P^*_{j\to i}; j \in pa(i)\}$ from all parents, while the latter involves the cost-to-go

function C_i^* expressed by (3.19), uniquely determined from the incoming backward messages $C^*_{ch(i)\to i} = \{C^*_{j\to i}; j \in ch(i)\}$ from all children. Thus, after substitution of (3.17) and (3.19), we see that the right-hand-side of (3.16) can be viewed as an operator $f_i(P^*_{pa(i)\to i}, C^*_{ch(i)\to i})$. Similarly, person-by-person optimality at every node other than i requires the outgoing messages from node i to its neighbors $pa(i) \cup ch(i)$. The outgoing forward messages $P^*_{i\to ch(i)} = \{P^*_{i\to j}; j \in ch(i)\}$ are collectively determined by the right-hand-side of (3.18), which after substitution of (3.17) and (3.14) we denote by the operator $g_i(\phi_i^*, P^*_{pa(i)\to i})$. The outgoing backward messages $C^*_{i\to pa(i)} = \{C^*_{i\to j}; j \in pa(i)\}$ are collectively determined by the right-hand-side of (3.20), which after substitution of (3.19) and (3.14) we denote by the operator $h_i(\phi_i^*, P^*_{pa(i)\to i}, C^*_{ch(i)\to i})$. Altogether, we see that Proposition 3.2 specializes the nonlinear fixed-point equations in (3.10) to the block-structured form

$$\begin{aligned} \phi_i &= f_i\left(P_{pa(i)\to i}, C_{ch(i)\to i}\right) \\ P_{i\to ch(i)} &= g_i\left(\phi_i, P_{pa(i)\to i}\right) \qquad\qquad i = 1, \ldots, n. \\ C_{i\to pa(i)} &= h_i\left(\phi_i, P_{pa(i)\to i}, C_{ch(i)\to i}\right) \end{aligned} \tag{3.21}$$

Corollary 3.3 (Offline Message-Passing Algorithm)**.** *Initialize all rule parameters* $\phi^0 = (\phi_1^0, \ldots, \phi_n^0)$ *and generate the sequence* $\{\phi^k\}$ *by iterating (3.21) in a repeated forward-backward pass through* $\mathcal{F}$ *e.g., iteration* $k = 1, 2, \ldots$ *is*

$$P^k_{i\to ch(i)} := g_i(\phi_i^{k-1}, P^k_{pa(i)\to i})$$

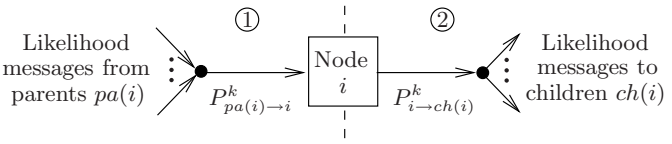
from $i = 1, 2, \ldots n$ *and*

$$\begin{aligned} \phi_i^k &:= f_i(P^k_{pa(i)\to i}, C^k_{ch(i)\to i}) \\ C^k_{i\to pa(i)} &:= h_i(\phi_i^k, P^k_{pa(i)\to i}, C^k_{ch(i)\to i}) \end{aligned}$$

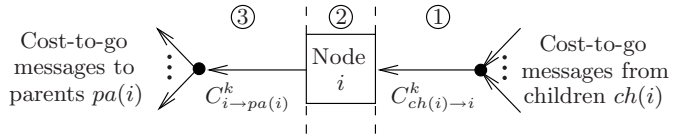
from $i = n, n-1, \ldots, 1$ *as illustrated in Figure 3.2. If Assumptions 3.2–3.4 hold, then the associated sequence* $\{J(\gamma^k)\}$ *converges.*

Proof. By virtue of Proposition 3.2, a sequence $\{\phi^k\}$ is the special case of a sequence $\{\theta^k\}$ considered in Corollary 3.1. Each forward-backward pass in the partial-order implied by $\mathcal{F}$ ensures each iterate ϕ^k is generated in the node-by-node coordinate descent fashion required for convergence. □

Proposition 3.2 also implies that, to carry out the iterations defined in Corollary 3.3, each node no longer needs a complete description of the global state distribution $p(x)$. This is arguably surprising, since we have not yet made a restrictive assumption about



(a) Forward Pass at Node i: "Receive & Transmit"



(b) Backward Pass at Node i: "Receive, Update & Transmit"

Figure 3.2. The distributed message-passing interpretation of the kth iteration in the offline algorithm discussed in Corollary 3.3, each node i interleaving its purely-local computations with only nearest-neighbor communications.

the state process X. As seen from (3.16)–(3.20), it is sufficient for each node i to know the joint distribution $p(x_{pa(i)}, x_i)$ of only the states local to itself and its parents. In our work here, we assume that these local probabilities are available at initialization. However, computing such local probabilities for a general random vector X has exponential complexity and must often be approximated. Of course, if process X is itself defined on a graphical model with tractable structure commensurate with the network topology $\mathcal{F}$, then the distributed computation to first obtain the local priors $p(x_{pa(i)}, x_i)$ at each node i is straightforward and tractable e.g., via belief propagation. For problems in which each node's local state X_i can also depend on its parents' decisions $U_{pa(i)}$, as considered in [81], Proposition 3.2 continues to apply provided we generalize the local prior available at each node i to the quantity $p(x_{pa(i)}, x_i|u_{pa(i)})$, then using it in place of the quantity $p(x_{pa(i)}, x_i)$ in (3.17) and (3.20).

A final implication of Proposition 3.2 is the simplicity with which the sequence $\{J(\gamma^k)\}$ can be computed. Specifically, the global penalty associated to iterate ϕ^k is given by

$$J(\gamma^k) := \sum_i G_i(\gamma^k)$$

with

$$G_i(\gamma^k) := \sum_{x_i} p(x_i) \sum_{u_i} \sum_{\hat{x}_i} c(u_i, \hat{x}_i, x_i) \sum_{z_i} P_i^{k+1}(z_i|x_i) p(u_i, \hat{x}_i|x_i, z_i; \phi_i^k)$$

for every i. That is, given that the likelihood function P_i^{k+1} is known local to each node i (which occurs upon completion of the forward pass in iteration $k+1$), each penalty

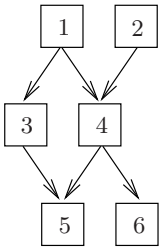
term G_i can be locally computed by each node i and, in turn, computation of the total penalty $J(\gamma^k)$ scales linearly in n.

As was the case for Corollary 3.1, the choice of initial parameter vector ϕ^0 in Corollary 3.3 can be important. Consider, for example, initializing to the myopic strategy $\bar{\delta} = (\bar{\delta}_1, \ldots, \bar{\delta}_n)$, where every node employs the rule in (2.4) that both ignores its received information and transmits no information (i.e., always transmits the same zero-cost symbol so that $J_c(\bar{\delta})$ is zero): given Assumption 3.2 and Assumption 3.3 both hold and also assuming

$$c(u_i, \hat{x}_i, x_i) = c(\hat{x}_i, x_i) + \lambda c(u_i, x_i)$$

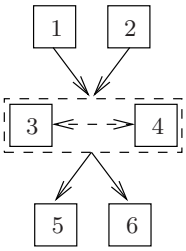
for every i, it turns out that this myopic strategy is person-by-person optimal! That is, the parameter vector $\phi = (\bar{\phi}_1, \ldots, \bar{\phi}_n)$ is itself a fixed-point of (3.21), and as such the algorithm will make no progress from the associated myopic (and typically sub-optimal) performance $J(\bar{\delta}) = J_d(\bar{\delta})$. While most details will vary for different classes of models, one general guideline is to initialize with a strategy such that every possible transmission/state pair (u_i, x_i) at every node i has a nonzero probability of occurrence. This will ensure that the algorithm explores, at least to some degree, the cost/benefit tradeoff of the online communication, making convergence to the myopic fixed-point likely only when λ is so large in (3.2) that communication penalty $J_c(\gamma^*)$ should be zero, as will be demonstrated by examples in Section 3.4.

Assumption 3.4 is arguably the most restrictive in Proposition 3.2, in the sense that satisfying it in practice must contend with non-local network connectivity constraints. For example, while any node may have more than one parent node, none of those parents may have a common ancestor. In principle, as illustrated in Figure 3.3, this restriction can be removed by merging such parent nodes together into single "super-nodes," but doing this recognizes the associated need for direct "offline" communication among these merged parent nodes while designing the decision rules (even though these decision rules continue to respect the online network topology $\mathcal{F}$). Combining such parent nodes also leads to increasing complexity in the offline computation local to that super-node (as we must consider the joint states/decisions at the nodes being merged); however, for sparse network structures, such merged state/decision spaces (if necessary) will still be of relatively small cardinality. Alternatively, there is nothing that prevents one from applying the message-passing algorithm as an efficient approximation within a general directed acyclic network, an idea we illustrate for a simple non-tree-structured model in Section 3.4.



$$p(u_3, u_4|x_3, x_4; \gamma) \neq P_{3\to 5}(u_3|x_3)P_{4\to 5}(u_4|x_4)$$
$$C_1(u_1, x_1) \neq C_{3\to 1}(u_1, x_1) + C_{4\to 1}(u_1, x_1)$$

(a) Non-tree structured network topology



$$p(u_3, u_4|x_3, x_4; \gamma) = P_{\{3,4\}\to 5}(u_3, u_4|x_3, x_4)$$
$$C_1(u_1, x_1) = C_{\{3,4\}\to 1}(u_1, x_1)$$

(b) Equivalent tree-structured network topology

Figure 3.3. An example of (a) a non-tree-structured network topology $\mathcal{F}$ and (b) its equivalent polytree topology for which Proposition 3.2 is applicable. Specifically, the parents of node 5, namely nodes 3 and 4, have node 1 as a common ancestor so we "merge" nodes 3 and 4. This is done at the (strictly offline) expense of requiring both direct communication between nodes 3 and 4 and increased local computation by nodes 3 and 4, so the message-passing algorithm in Corollary 3.3 can jointly consider the random variables X_3, X_4, U_3 and U_4.

■ 3.4 Examples and Experiments

This section summarizes experiments with the offline message-passing algorithm presented in Section 3.3. Throughout, as will be detailed in Subsection 3.4.1, we model each sensing node as the linear binary detector of Example 2.4 and each communication link as the peer-to-peer erasure channel of Example 3.1. We define the global costs $c(\hat{x}, x)$ and $c(u, x)$ so that detection penalty J_d and communication penalty J_c measure precisely the gateway node-error-rate and network-wide link-use-rate, respectively. Our purpose is to characterize the team-optimal performance, examining the tradeoff formulated in Section 3.2 relative to the benchmark centralized and myopic solutions

discussed in Chapter 2. Our procedure is to sample the range of λ in (3.2), each time recording the point (J_d, J_c) achieved by the message-passing algorithm.

Subsections 3.4.2–3.4.4 present experimental results across different network topologies, different levels of measurement/channel noise and different prior probability models. These results illustrate how the decentralized strategy produced by the message-passing algorithm consistently exploits the selective transmission scheme: even when actual symbols can be transmitted reliably and without penalty (i.e., when erasure probabilities are zero and $\lambda = 0$ in (3.2)), a node's selective silence can convey valuable information to its children. Our experimental procedure also records the average number of message-passing iterations to convergence, recognizing that per offline iteration k each link (i, j) must reliably compute and communicate messages $P^k_{i\to j}$ and $C^k_{j\to i}$, each a collection of up to $|\mathcal{X}_i \times \mathcal{U}_i|$ real numbers. This empirical measure of offline overhead is, we believe, an important point in understanding the value and feasibility of self-organizing sensor networks, as it allows us to assess the price of adaptive organization, or re-organization. In particular, our analysis emphasizes that for such offline organization to be warranted, it must be that the price of performing it can be amortized over a substantial number of online usages, or equivalently that the network resources consumed for organization represent only a modest fraction of the resources available over the total operational lifetime.

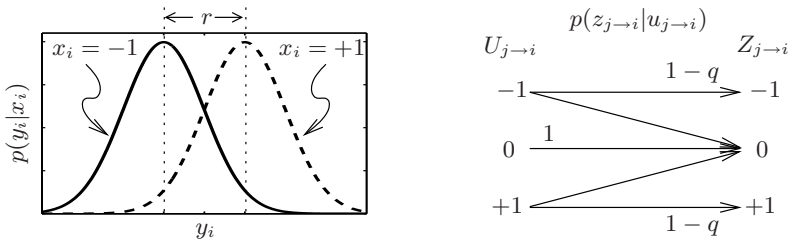
■ 3.4.1 Local Node Models

To apply the offline message-passing algorithm developed in Section 3.3 given a directed network topology $\mathcal{F}$, each node i requires the following local models: likelihoods $p(y_i|x_i)$, channels $p(z_i|x_i, u_{pa(i)})$, costs $c(u_i, \hat{x}_i, x_i)$, priors $p(x_{pa(i)}, x_i)$ and an initial rule $\gamma^0_i \in \Gamma_i$. This subsection describes the parametric forms of the local models that are in common with all experiments to be described in the following subsections. In particular, only the local priors will be different across these experiments, so we now describe all other such local models and leave the description of priors for later subsections.

The global sensing model is that of the n independent linear Gaussian binary detectors as introduced in Example 2.8, assuming homogeneous sensors i.e., $r_i \equiv r$ for all i. We restate this sensing model here for convenience: each node's local likelihood is given by

$$p(y_i|x_i) = \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{1}{2}\left(y_i - \frac{rx_i}{2}\right)^2\right), \qquad (x_i, y_i) \in \{-1, +1\} \times \mathbb{R}$$

with parameter $r \in (0, \infty)$ inversely related to the measurement noise level local to



(a) Measurement Model at Node i (b) Channel Model at Link (j,i)

Figure 3.4. The per-node measurement model and per-link channel model used in our experiments: (a) the ith node's likelihood function $p(y_i|x_i)$, defining a linear-Gaussian binary detector with parameter $r \in (0,\infty)$ inversely related to the measurement noise level; and (b) the transition probabilities defining the point-to-point link to node i from each parent $j \in pa(i)$, each such link (j,i) with parameter $q \in [0,1]$ directly related to the channel noise level. Even though node j selecting $U_{j\to i} = 0$ avoids the potential of a link erasure, upon receiving $Z_{j\to i} = 0$, node i will not be able to determine conclusively (unless $q = 0$) whether parent j elected to be silent or to transmit an actual symbol but link (j,i) then experienced an erasure.

each node; see Figure 3.4(a). The myopic rule in (2.4) then reduces to a threshold test, where parameters $\bar{\phi}(\hat{x}_i, x_i) = p(x_i)c(\hat{x}_i, x_i)$ collectively determine the myopic threshold,

$$\frac{p_{Y_i|X_i}(y_i|+1)}{p_{Y_i|X_i}(y_i|-1)} = \exp(ry_i) \;\equiv\; \Lambda_i(y_i) \;\begin{array}{c}\hat{x}_i=+1\\ >\\ <\\ \hat{x}_i=-1\end{array}\; \bar{\eta}_i \;\equiv\; \frac{\bar{\phi}_i(+1,-1)-\bar{\phi}_i(-1,-1)}{\bar{\phi}_i(-1,+1)-\bar{\phi}_i(+1,+1)}.$$

In turn, the local marginalization over Y_i reduces to computing the false-alarm and true-detection probabilities, for any fixed threshold value η_i given by

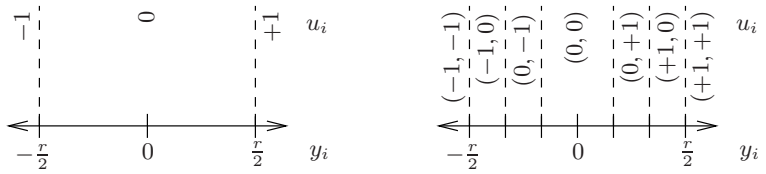
$$p_{\hat{X}_i|X_i}(+1|x_i) = \int_{\log(\eta_i)/r}^{\infty} p(y_i|x_i)\, dy_i$$

when $x_i = -1$ and $x_i = +1$, respectively.

The channel model local to each node i assumes all incoming links from parents $pa(i)$ are mutually-independent erasure channels as introduced in Example 3.1, each also independent of the local state process X_i. More specifically, we assume

$$p(z_i|x_i, u_{pa(i)}) = \prod_{j\in pa(i)} p(z_{j\to i}|u_{j\to i})$$

where for each link (j,i), as depicted in Figure 3.4(b), both the transmitted symbol $U_{j\to i}$ and the received symbol $Z_{j\to i}$ take values in the ternary alphabet $\{-1,0,+1\}$ and parameter $q \in [0,1]$ is equal to the link's erasure probability. As was discussed in



(a) Given Node i has One Child (b) Given Node i has Two Children

Figure 3.5. The initial rule γ_i^0 used in our experiments. For all z_i, we partition the real y_i-axis into a set of intervals with $2(2^{|ch(i)|}-1)$ thresholds to decide $u_i \in \{-1,0,+1\}^{|ch(i)|}$ and a threshold of zero to decide $\hat{x}_i \in \{-1,+1\}$. In essence, each node i is initialized to (i) ignore all information received on the incoming links, (ii) myopically make a maximum-likelihood estimate of its local state and (iii) make a binary-valued decision per outgoing link (i,j), remaining silent (with $u_{i\to j}=0$) when the measurement is near its least-informative values or transmitting its local state estimate (with $u_{i\to j}=\hat{x}_i$) otherwise.

Example 3.3, the event $U_{j\to i}=0$ represents node j suppressing its transmission to child $i \in ch(j)$, in which case the event $Z_{j\to i}=0$ occurs with probability one: however, each actual transmission by parent $j \in pa(i)$ (represented by the event $U_{j\to i}\neq 0$) is erased (represented by $Z_{j\to i}=0$) with probability q or otherwise successfully received by node i (represented by $Z_{j\to i}=U_{j\to i}$).

The cost function local to node i is defined such that detection penalty J_d and communication penalty J_c in (3.2) equal the gateway *node-error-rate* and network-wide *link-use-rate*, respectively. Specifically, letting $g(i)=1$ denote that node i is in the gateway and $g(i)=0$ denote otherwise, the global cost function satisfies Assumption 3.3 with each ith term given by

$$c(u_i,\hat{x}_i,x_i) = g(i)c(\hat{x}_i,x_i) + \lambda \sum_{j\in ch(i)} c(u_{i\to j}),$$

where the detection-related costs indicate node errors and the communication-related costs indicate link uses i.e.,

$$c(\hat{x}_i,x_i)=\begin{cases}0, & \hat{x}_i=x_i\\ 1, & \hat{x}_i\neq x_i\end{cases} \quad \text{and} \quad c(u_{i\to j})=\begin{cases}0, & u_{i\to j}=0\\ 1, & u_{i\to j}\neq 0\end{cases}.$$

As discussed for the channel model in Figure 3.4, the event $U_{i\to j}=0$ indicates that node i suppresses the transmission on the outgoing link to child j, so it is associated to zero communication cost. Also note that the myopic threshold for each gateway node reduces to $\bar{\eta}_i = p_{X_i}(-1)/p_{X_i}(+1)$.

A final consideration in the model local to node i is the initial rule γ_i^0. As remarked after Corollary 3.3 in Section 3.3, initializing to a myopic rule in (2.4) would prohibit the offline algorithm from making progress. Figure 3.5 illustrates our choice of initial

rule γ_i^0, and we observe the algorithm making reliable progress from this initialization as long as the induced statistics at every node i satisfy $p(u_i|x_i;\gamma_i^0) > 0$ for all $(x_i, u_i) \in \{-1,+1\} \times \{-1,0,+1\}^{|ch(i)|}$. In the absence of parents, this rule is equivalent to the class of monotone threshold rules for linear-Gaussian binary detectors described in Example 2.5. The same threshold parameterization extends to nodes with parents, only that there can be $|\mathcal{Z}_i|$ such partitions of the likelihood-ratio space $[0,\infty)$, namely one set of such thresholds per symbol value $Z_i = z_i$.

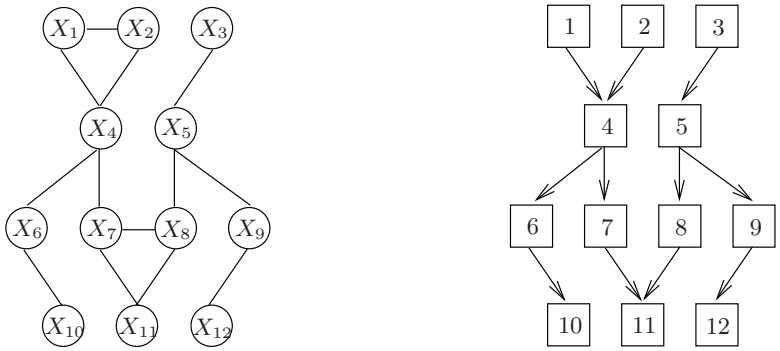
■ 3.4.2 A Small Illustrative Network

This subsection assumes the local models discussed in the preceding subsection, and considers the prior probability model $p(x)$ and network topology $\mathcal{F}$ depicted in Figure 3.6. Specifically, let the hidden state process X be Markov on the undirected graph $\mathcal{G}$ illustrated in Figure 3.6(a), assuming edge potentials

$$\psi(x_i, x_j) = \begin{cases} w & , \quad x_i = x_j \\ 1-w & , \quad x_i \neq x_j \end{cases}$$

that express the correlation (i.e., negative, zero, or positive when w is less than, equal to, or greater than 0.5, respectively) between neighboring binary-valued states X_i and X_j. Note that, with just $n = 12$ nodes, the computation to obtain the neighborhood marginal $p(x_{\pi(i)}, x_i)$ for each node i can be performed directly. Also observe that, in this example, the links in the network topology are a proper subset of the edges in the (loopy) undirected graph upon which X is defined.

Figure 3.7 displays the tradeoff between node-error-rate J_d and link-use-rate J_c achieved by the message-passing algorithm across different model parameters. In each case, every node is in the gateway (so that the maximal node error rate is twelve) and, for each parameter pair (w, r) under investigation, the tradeoff curve is computed for three different erasure probabilities. We see that these three curves always start from a common point, corresponding to λ being large enough so that zero link-use-rate (and thus myopic node-error-rate) is optimal. The smallest value of λ achieving this myopic point, call it λ^*, can be interpreted (for that model instance) as the maximal price (in units of detection penalty) that the optimized network is willing to pay per unit of online communication. For λ less than λ^*, we see that the message-passing algorithm smoothly trades off increasing link-use-rate with decreasing node-error-rate. Not surprisingly, this tradeoff is most pronounced when the erasure probability q is zero, and approaches the myopic detection penalty as q approaches unity. Also shown per instance of parameters



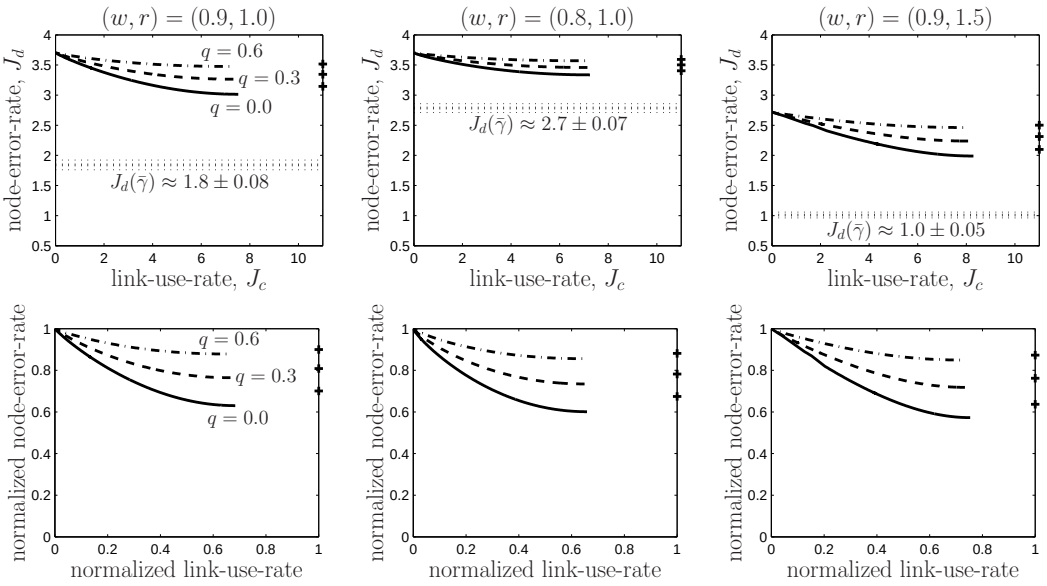
(a) Correlated Binary States (b) Directed Polytree Network

Figure 3.6. A small ($n = 12$) decentralized detection network used in our experiments: (a) the (undirected) graph $\mathcal{G}$ upon which the spatially-distributed state process X is defined and (b) a tree-structured (directed) network topology $\mathcal{F}$ that spans the vertices in (a). Observe that the links in the polytree network topology in (b) are a proper subset of the edges in the undirected graph in (a).

(w, r) is a Monte-Carlo estimate of the optimal centralized performance $J_d(\bar{\gamma})$, computed using 1000 samples from $p(x, y)$ and simulating the strategy in (2.2).

The second row of curves displays the same data as in the first row, but after (i) normalizing the achieved link-use-rate by its capacity (i.e., eleven unit-rate links) and (ii) expressing the achieved node-error-rate on a unit-scale relative to the benchmark centralized detection penalty and the myopic detection penalty (i.e., representing the fraction of this centralized versus myopic gap gained via team-optimized coordination). These rescalings emphasize that the maximum link-use-rates on each optimized curve are well below network capacity and that the message-passing algorithm consistently converges to a strategy that exploits the selective silence: intuitively, each node in the cooperative strategy is able to interpret "no news as providing news." The curves show that, subject to less than eleven bits (per global estimate) of online communication, up to 40% of the optimal performance lost by the purely myopic strategy can be recovered. For further comparison, consider the model with selective communication disabled, meaning each node must always transmit either a $+1$ or -1 to each of its children and, in turn, link-use-rate is at 100% capacity. Applying the message-passing algorithm to these models yields the points indicated by "+" marks: indeed, we see that selective communication affords up to an additional 10% recovery of detection performance while using only 70% of the online communication capacity.

The tables in Figure 3.7 list two key quantities recorded during the generation of each of the nine tradeoff curves, namely λ^* and k^* denoting the lowest value of λ for which



q	λ^*	k^*
0.0	0.2061	3.2 ± 0.44
0.3	0.1557	3.0 ± 0.17
0.6	0.0960	2.8 ± 0.42

(a) Nominal Environment

q	λ^*	k^*
0.0	0.1494	3.0 ± 0.04
0.3	0.1119	2.9 ± 0.27
0.6	0.0684	2.5 ± 0.50

(b) Low State Correlation

q	λ^*	k^*
0.0	0.1356	3.3 ± 0.78
0.3	0.0963	3.1 ± 0.33
0.6	0.0633	3.0 ± 0.08

(c) Low Measurement Noise

Figure 3.7. Optimized tradeoff curves for the model in Subsection 3.4.2 given (a) a nominal environment, (b) low state correlation and (c) low measurement noise, each such environment with three different link erasure probabilities $q = 0$ (solid line), 0.3 (dashed line) and 0.6 (dash-dotted line). Each curve is obtained by sampling λ in increments of 10^{-4}, starting with $\lambda = 0$, and declaring convergence in iteration k when $J(\gamma^{k-1}) - J(\gamma^k) < 10^{-3}$. The second row of figures uses the same data as the first, normalizing the two penalties to better compare across the different model instances. The tables contain the two key quantities λ^* and k^* we record while computing each curve, respectively the lowest value of λ for which the myopic operating point is team-optimal and the average number of offline iterations to convergence. See Subsection 3.4.2 for more discussion of these results.

the myopic point is optimal and the average number of offline iterations to convergence, respectively. As discussed above, the former can be interpreted as the "fair" per-unit price of online communication: indeed, from the tables, we see that λ^* is inversely related to erasure probability q, quantifying the diminishing value of active transmission as link reliability degrades. Moreover, comparing λ^* in (a) with those in (b) and (c), we see that lower state correlation or lower measurement noise similarly diminish the value of active transmission. The empirical value of k^* is related to the price of offline

self-organization: we see that it measures between 3 and 4 iterations, implying that maintaining the optimized online tradeoff depends (per offline reorganization) upon the exact computation and reliable communication of roughly 684 to 912 real numbers in total, or roughly 57 to 76 real numbers per node.

■ 3.4.3 Large Randomly-Generated Networks

This subsection performs a similar analysis as in Subsection 3.4.2, except that we consider a collection of randomly-generated model instances of more realistic size and character. Figure 3.8 illustrates a typical output of our model generation procedure: it starts with $n = 100$ nodes, each randomly positioned within a unit-area square and connected to a randomly selected subset of its spatial neighbors. The vector state process X is this time described by a directed graphical model, constructed such that the correlation between neighboring states reflects the spatial proximity of the neighbors; specifically, we let $d(i,j)$ be the spatial distance between node i and node j and, denoting by $\bar{pa}(i)$ the parents of each node i on the probability graph $\mathcal{G}$, we choose

$$p(x_i|x_{\bar{pa}(i)}) = \begin{cases} 1-\rho(x_{\bar{pa}(i)}) & , \quad x_i = -1 \\ \rho(x_{\bar{pa}(i)}) & , \quad x_i = +1 \end{cases},$$

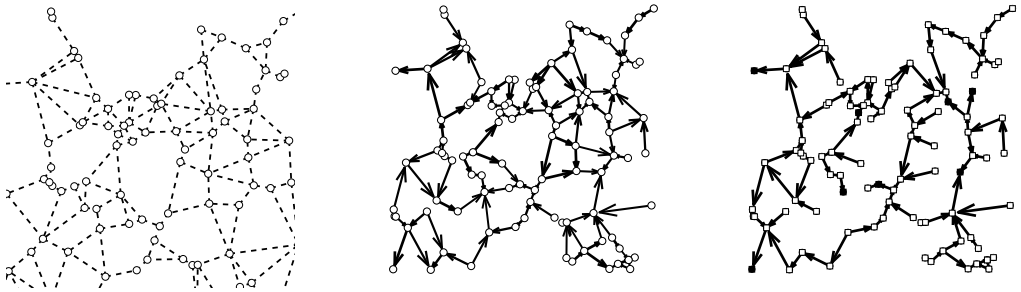
$$\rho(x_{\bar{pa}(i)}) = \frac{\sum_{j \in \bar{pa}(i)} 1_j(x_{\bar{pa}(i)}) d(i,j)^{-1}}{\sum_{j \in \bar{pa}(i)} d(i,j)^{-1}},$$

$$1_j(x_{\bar{pa}(i)}) = \begin{cases} 1 & , \quad x_j = +1 \\ 0 & , \quad x_j = -1 \end{cases}.$$

Given such a directed graphical model for X, we use Murphy's *Bayesian Network Toolbox in Matlab* [69]) to find the clique marginals $p(x_{\bar{pa}(i)}, x_i)$ for each i. Note that, further exploiting the Markov properties of X, this allows us to readily compute the neighborhood marginals (for the probability graph $\mathcal{G}$) via

$$p(x_{\bar{ne}(i)}, x_i) = p(x_{\bar{pa}(i)}, x_i, x_{\bar{ch}(i)}) = p(x_{\bar{pa}(i)}, x_i) \prod_{j \in \bar{ch}(i)} p(x_j|x_i). \tag{3.22}$$

The next step of model generation is to select ten gateway nodes at random, which in these particular experiments we assume will be the childless nodes of a spanning polytree network $\mathcal{F}$. We then build this network via Kruskal's spanning tree algorithm, maximizing edge weights proportional to the pairwise correlation between the states



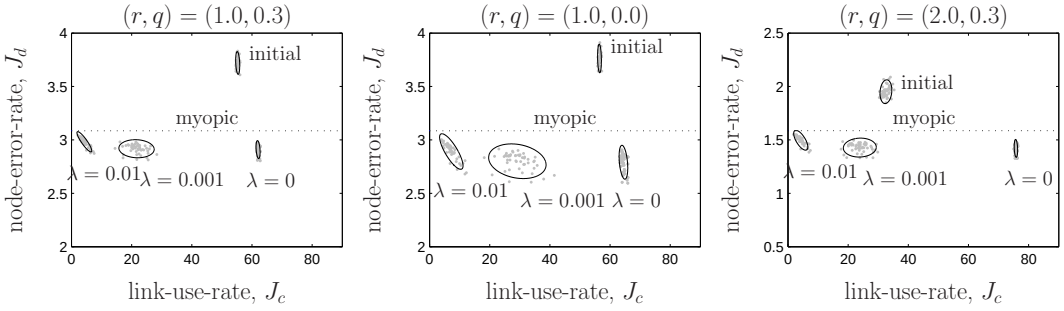
(a) Random Spatial Configuration (b) Directed Graphical Model (c) Polytree Network Topology

Figure 3.8. A typical 100-node detection network generated randomly for our experiments: (a) the spatial configuration of all nodes in the unit-area square, (b) an arbitrary directed acyclic graph $\mathcal{G}$ upon which the spatially-distributed state process X is defined and (c) the polytree network topology $\mathcal{F}$, where the ten randomly-selected gateway nodes are denoted by the filled node markers. Subsection 3.4.3 described the construction in more detail.

sharing each edge. Thus, the directed polytree $\mathcal{F}$ has a topology contained in the undirected topology of $\mathcal{G}$ (i.e., $pa(i) \subseteq \bar{ne}(i)$ for every i, where $pa(i)$ denote the parents on the communication graph $\mathcal{F}$) and so the local marginals $p(x_{pa(i)}, x_i)$ for every node i required by the offline message-passing algorithm can be found by appropriate marginalization of (3.22).

Figure 3.9 depicts the average-case performance achieved by the message-passing algorithm over 50 randomly-generated model instances. Each plot consists of four clusters of points, three corresponding to the optimized point assuming three different values of λ and one corresponding to the point achieved by a heuristic strategy, which essentially interprets each incoming symbol as indicating the true value of the neighbors' local states. We see that the heuristic strategy fails catastrophically, in the sense that communication penalty is nonzero and yet the detection penalty is larger than even that of the myopic strategy! This unsatisfactory heuristic performance underscores the value of our offline message-passing algorithm, which via parameter λ consistently decreases global detection penalty (from that of the myopic strategy) as global communication penalty increases.

Also shown for each optimized cluster is k^*, or the average number of iterations to convergence, which underscores the price of our offline coordination in the same sense discussed in Subsection 3.4.2. We see that roughly eight iterations can be required in the 100-node models, in comparison to roughly three iterations in the twelve-node models of the previous subsection, suggesting the price of offline coordination scales sublinearly with the number of nodes n. It is worth noting that the communication



λ	k^*
0.000	5.2 ± 0.40
0.001	5.2 ± 0.47
0.010	4.8 ± 0.61

(a) Nominal Environment

λ	k^*
0.000	7.8 ± 0.98
0.001	8.4 ± 0.93
0.010	7.8 ± 1.34

(b) Zero Channel Noise

λ	k^*
0.000	4.1 ± 0.32
0.001	4.0 ± 0.35
0.010	4.9 ± 0.96

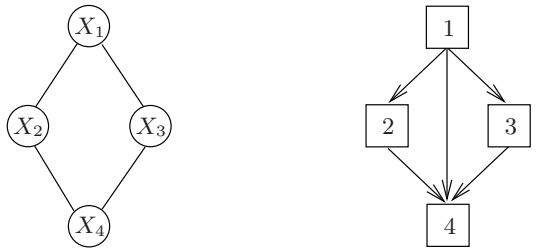
(c) Low Measurement Noise

Figure 3.9. Performance of five different strategies for 50 randomly generated models of the type described in Subsection 3.4.3 given (a) a nominal environment, (b) zero channel noise and (c) low measurement noise. In each plot, the dotted horizontal line is the detection penalty achieved by the myopic strategy; the three clusters below this dotted line shows the performance of the optimized strategies for three different values of λ, and the cluster above the myopic strategy shows the performance of a heuristic strategy. Each ellipse is the least-squares fit to the 50 data points associated to each candidate strategy. For the three optimized strategies, we declare convergence in iteration k when $J(\gamma^{k-1}) - J(\gamma^k) < 10^{-3}$, and each table lists the average number of offline iterations to convergence. See Subsection 3.4.3 for more discussion of these results.

overhead associated with each offline iteration also depends on the connectivity of the network topology, each node exchanging a number of messages that scales linearly with its degree.

■ 3.4.4 A Small Non-Tree-Structured Network

The preceding experiments focused on models that satisfy all assumptions under which the offline message-passing algorithm is derived. We now discuss experiments for a model in which the network topology is *not* a polytree. In such cases the local fixed-point equations in Corollary 3.3 are no longer guaranteed to be equivalent to the general fixed-point equations in Corollary 3.1. In turn, the message-passing algorithm no longer necessarily inherits the general convergence and correctness guarantees discussed for Corollary 3.1. As remarked in Section 3.3, the team-optimal solution can be computed by aggregating nodes in the original graph so as to form a polytree to which our message-passing algorithm can be applied. Of course, such a process implicitly requires



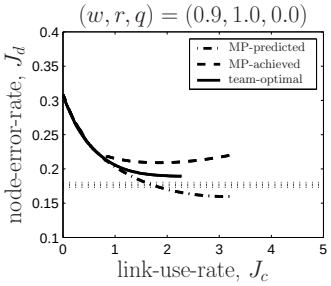
(a) Correlated Binary States (b) Directed Non-Tree Topology

Figure 3.10. Another small ($n = 4$) decentralized detection network used in our experiments: (a) the (undirected) graph upon which the spatially-distributed state process X is defined and (b) the (directed) network topology that spans the vertices in (a). Observe that, in contrast to our preceding experiments, the network topology (i) is *not* tree-structured and (ii) includes a link between two nodes that do not share an edge in (a).

communication among nodes that have been aggregated but are not neighbors in the original graph. Moreover, this approach is computationally tractable only if a small number of nodes need to be aggregated.

For the above reasons, it is useful to understand both what the fully team-optimal methods can achieve as well as what can be accomplished if we simply apply the local message-passing algorithm to the original non-tree-structured graph. In this section, we present and discuss experiments on a small example in order to explore these questions. Even in such small models, the team-optimal solution is seen to produce rather sophisticated signaling strategies, exploiting the non-tree network structure in ways that cannot be accomplished via the message-passing approximation. Nonetheless, with regard to achieved performance, our analysis of these simple models suggests that the local message-passing algorithm can provide an effective approximation.

Let us consider a model of the same type as in Subsection 3.4.2, except involving only four nodes in the non-tree configuration depicted in Figure 3.10. Assume for illustration that $r = 1$, $q = 0$, and $w = 0.9$, so that all measurements have the same noise, all channels have zero erasure probability and the states are (attractively) Markov on the single-cycle graph in Figure 3.10(a). Moreover, assume node 4 is the lone gateway node, while nodes 1, 2 and 3 are communication-only nodes. The team objective essentially boils down to having the communication-only nodes collectively generate the "most-informative-yet-resourceful" *signal* to support the gateway node's final decision. Indeed, we should anticipate node 1 to play the dominant role in any such signaling strategy, given its direct link to every other node in the communication network topology of Figure 3.10(b). Note, in particular, that this communication topology includes a direct



solution method	λ^*	k^*
message-passing	0.145	3.2 ± 0.33
team-optimal	0.192	3.8 ± 0.46

Figure 3.11. Performance comparison between the team-optimal solution and the message-passing approximation for the non-tree-structured model in Subsection 3.4.4. Three tradeoff curves are shown, dashed being that achieved by the message-passing approximation, solid being that achieved by the team-optimal solution, and dash-dotted being that predicted by the message-passing approximation. Each curve is obtained by sampling λ in increments of 10^{-3}, starting with $\lambda = 0$, and declaring convergence in iteration k when $J(\gamma^{k-1}) - J(\gamma^k) < 10^{-3}$. Also shown is the empirical estimate (plus or minus one standard deviation based on 10000 samples) of the optimal centralized performance. See Subsection 3.4.4 for more discussion of these results.

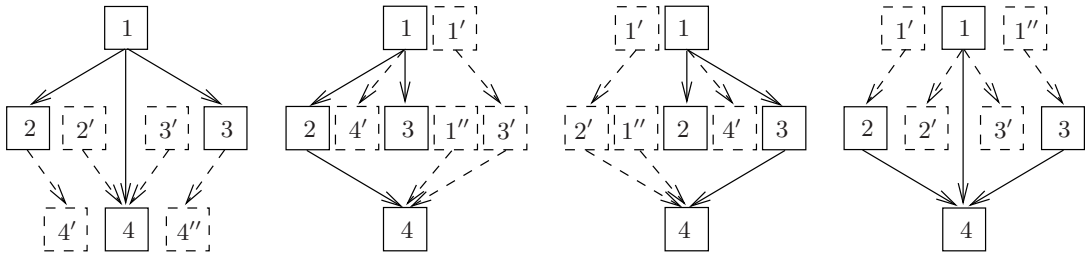
path from node 1 to node 4 which is not present in the graph of Figure 3.10(a) which captures the statistical structure among the variables sensed at each node. Thus, this example also allows us to illustrate the value of longer-distance messaging than would be found, for example, if loopy belief propagation were applied to this problem.

Figure 3.11 displays the tradeoff between node-error-rate J_d and link-use-rate J_c achieved by both the team-optimal solution and the message-passing approximation. We also show the performance tradeoff *predicted* by the message-passing algorithm (but based on incorrect assumptions). All three curves coincide at very low link-use-rates, a regime in which enough links remain unused so that the network topology is effectively tree-structured. For higher link-use-rates, we see that the message-passing prediction is consistently over-optimistic, eventually even suggesting that the achieved node-error-rate surpasses the optimal centralized performance in the actual network; meanwhile, the actual performance achieved by the message-passing approximation is consistently inferior to that of the team-optimal solution, yet for this simple model still a reliable improvement relative to myopic detection performance. Also notice how the message-passing approximation does not produce a monotonic tradeoff curve, in the sense that it permits link-use-rates to increase beyond the range over which the node-

error-rate remains non-increasing. The team-optimal solution is, of course, monotonic in this sense, with peak link-use-rate well below that determined by the message-passing approximation. Finally, the table in Figure 3.11 shows that the team-optimal solution is (i) more resourceful with its link usage, as quantified by λ^*, and (ii) takes on-average more iterations to converge, as quantified by k^*. The latter is arguably surprising, considering it is the message-passing approximation that comes without any theoretical guarantee of convergence. Indeed, these particular experiments did not encounter a problem instance in which the message-passing algorithm failed to converge.

We conjecture that algorithm convergence failures will be experienced when the message-passing approximation is applied to more elaborate non-tree-structured models. To help justify this point, Figure 3.12 depicts the key discrepancy between the team-optimal solution and the message-passing approximation. As each node performs each of its local message-passing iterations, it neglects the possibility that any two parents could have a common ancestor (or, equivalently, that any two children could have a common descendant), implicitly introducing fictitious replications of any such neighbors and essentially "double-counting" their influence. This replication is reminiscent of the replications seen in the so-called *computation tree* interpretation of loopy belief propagation [100]. However, there are important differences in our case, as this replication is both in upstream nodes that provide information to a specific node *and* in downstream nodes whose decision costs must be propagated back to the node in question. Moreover, the nature of these replications is itself node-dependent, meaning each iteration of the algorithm may be cycling over n different assumptions about the global network structure.

The potential for erroneous message iterates illustrated in Figure 3.12 manifests itself in the performance difference, most apparent for small values of λ, between the solutions compared in Figure 3.11. While both solutions yield a signaling strategy in which node 1 takes a leadership role, the team-optimal strategy consistently uses nodes 2 and 3 in a more resourceful way, ultimately allowing gateway node 4 to receive better side information for its final decision. We have more carefully explored this claim by considering plots of the type depicted in Figure 3.13, concluding the following. Firstly, in the team-optimal solution, node 1 typically signals exclusively to node 4 or exclusively to node 3, and only for the most discriminative local measurement will it signal to both nodes 2 and node 4; that is, node 1 never signals all three other nodes and, moreover, the signaling rules used by nodes 2 and 3 are asymmetric. In the message-passing approximation, however, node 1 typically uses either none or all of its links, in the



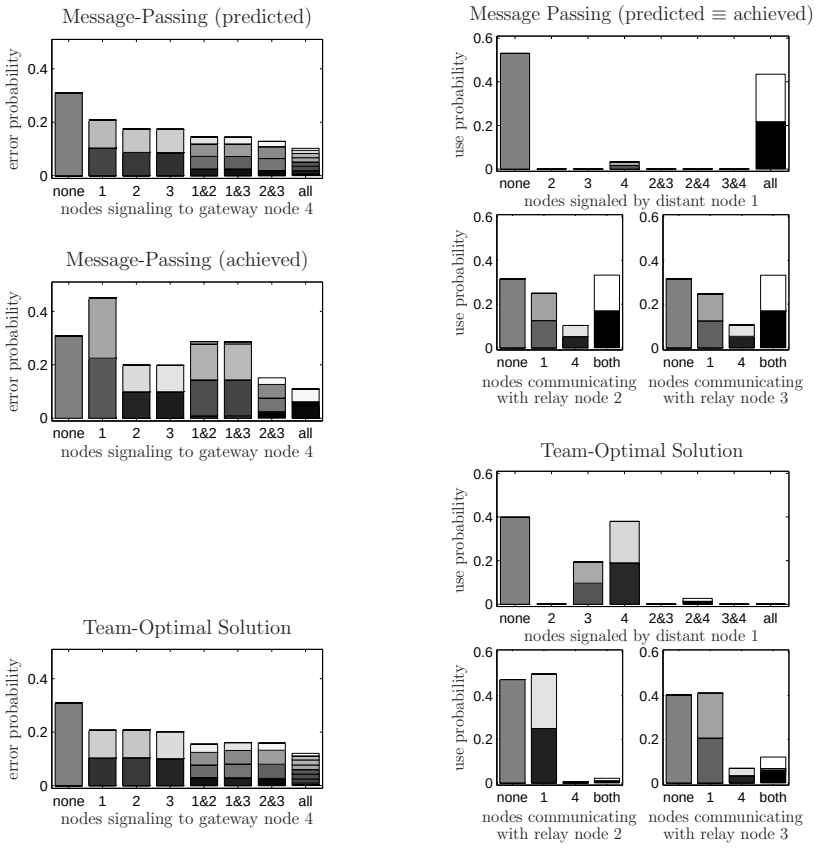
(a) Node 1 Perspective (b) Node 2 Perspective (c) Node 3 Perspective (d) Node 4 Perspective

Figure 3.12. The tree-based message-passing approximation from the perspective of each node in the non-tree structured model of Figure 3.10. Nodes and links drawn with dashed lines represent the fictitious nodes introduced by the approximation, which neglects the possibility that any two parents could have a common ancestor (or, equivalently, any two children could have a common descendent). The potential for these different perspectives to give rise to erroneous message iterates lies at the heart of the possibility for convergence difficulties in more elaborate non-tree-structured models.

latter case transmitting the same symbol to all other nodes; in turn, nodes 2 and 3 employ identical signaling rules to node 4 in which, given node 1 has communicated, the presence or absence of signal indicates agreement or disagreement, respectively, with the symbol broadcasted by node 1. In short, the message-passing approximation cannot recognize the value of introducing asymmetry and, consequently, determines that a larger network-wide link-use-rate is necessary to achieve a comparable gateway node-error-rate. A final observation is that the actual link-use probabilities achieved by the signaling rules of nodes 1,2 and 3 match those predicted by the message-passing approximation, reflecting how (in this example) the tree-based assumption is violated only once the fusion rule of gateway node 4 enters the picture.

■ 3.5 Discussion

This chapter presented our first inroads into addressing a key challenge in modern sensor networks, namely the inherent design tradeoffs between maximizing application-layer decision performance (e.g, node-error-rate) and maximizing network-layer energy efficiency (e.g., link-use-rate). Assuming a decision architecture based on only a single forward sweep in a directed acyclic network, we were able to heavily leverage known results of the well-studied decentralized detection paradigm. Mitigating performance loss in the presence of such severe online resource constraints demands an offline "self-organization" algorithm by which the processing rules local to all nodes are iteratively coupled in a manner driven by global problem statistics. We contributed to this body



(a) Fusion Rule of Node 4 (b) Signaling Rules of Nodes 1, 2 and 3

Figure 3.13. The different decentralized processing strategies found by the message-passing approximation and the team-optimal solution for the non-tree-structured model in Subsection 3.4.4 when $\lambda = 0.01$. Each strategy is comprised of (a) the fusion rule at gateway node 4 and (b) the signaling rules at communication-only nodes 1,2 and 3. The gray-scales in (a) indicate how the different incoming signals encode different regions of the likelihood function $p(z_4|x_4)$ that, ultimately, biases the gateway's local processing of measurement Y_4, with entirely gray meaning no bias (as when "none" of the nodes signal) and darker or lighter shades meaning a greater bias towards $X_4 = -1$ or $X_4 = +1$, respectively. For instance, consider the bars shown when "all" nodes signal to the gateway: the fusion rule achieved by the message-passing approximation is seen to almost always output the decision that agrees with two of the three incoming signals; in contrast, the team-optimal solution is seen to only be so heavily biased when all incoming signals agree, and otherwise counts each incoming signal with essentially equal weight. The gray scales in (b) indicate, for each communication-only node $i \in \{1, 2, 3\}$, how the different local signals encode different regions of the composite likelihood function $p(u_i, z_i|x_i)$, with gray denoting a likelihood near unity (as when "none" of the nodes are signaling) and darker or lighter shades denoting belief in favor of $X_i = -1$ or $X_i = +1$, respectively. For instance, in both solutions, node 1 maps only its most discriminative likelihood values into the decision to signal multiple nodes.

of research by showing that, for a certain class of models, this offline algorithm admits an efficient message-passing interpretation: it can be implemented as a sequence of purely-local computations interleaved with only nearest-neighbor communications. Our experiments with the efficient message-passing algorithm underscored how: (i) the algorithm can produce very resourceful cooperative processing strategies in which each node becomes capable of using the absence of communication as an additional informative signal; (ii) design decisions to reduce online resource overhead by imposing explicit in-network processing constraints must be balanced with the offline resource expenditure to optimize performance subject to such constraints; and (iii) the message-passing algorithm can be successfully applied to models that do not necessarily satisfy all of the assumptions under which it is originally derived.

Inherent to the single-sweep directed architecture considered here is that nodes with few ancestors are unlikely to make reliable state-related decisions in comparison to those with more ancestors (i.e., nodes at the "end of the line" access more global side information). Moreover, a directed architecture may not be easily compatible with emerging concepts in ad-hoc networking, as enforcing a directed acyclic topology on the fly could necessitate expensive non-local network-layer coordination among the distributed nodes. These issues motivate the consideration of a less constraining decision architecture, allowing for bidirectional inter-sensor communication defined on an *undirected* network topology. In the next chapter, we focus on the simplest such online processing architecture, analogous to running exactly one parallel iteration of belief propagation (per global measurement) with the same network-constrained twists (e.g., finite-alphabet messages) considered in this chapter. In Chapter 5, we carry this analogy even further, considering decision architectures built upon repeated forward-backward sweeps in a directed network and multiple parallel iterations on undirected networks. The connection between designing local decision rules and modifying factors of the conditional distribution $p(u, \hat{x}|x)$, already emphasized in this chapter, will be seen to play an increasingly important role.

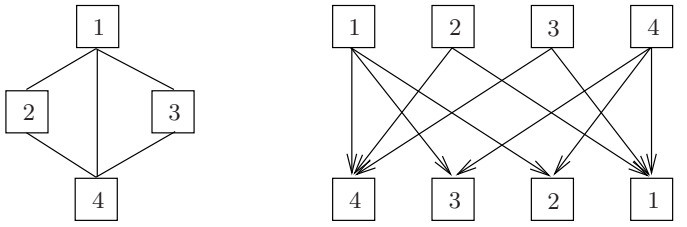
Chapter 4

Undirected Network Constraints

THIS chapter begins our departure from the mainstream decentralized detection literature, which focuses almost exclusively on unidirectional inter-sensor communication defined on a directed graph, by considering a non-ideal communication model defined on an *undirected* graph. Each edge in this graph is taken to indicate a bidirectional (and perhaps unreliable) finite-rate communication link between two distributed sensor nodes. An undirected network topology is arguably more compatible with the vision of wireless sensor networks discussed in Chapter 1, since enforcing a directed acyclic network topology "on the fly" may require expensive non-local coordination among the distributed nodes. Moreover, if the online message schedule is restricted to a single unidirectional sweep through the network, then only the nodes towards the end of the forward partial-order are afforded the opportunity to make "globally-aware" estimates of their local states. While the simplest directed architecture considered in the previous chapter may be satisfactory if final decisions are to be made at a comparatively small set of "fusion centers," other applications may desire quality state estimates at many or all nodes of the network (as was assumed in Subsection 3.4.2, for example, in which all nodes were gateway nodes).

■ 4.1 Chapter Overview

The initial focus in this chapter is to adapt the Bayesian detection formulation and team-theoretic analysis in Chapter 3 for a simplest undirected communication architecture, constraining the online message schedule to exactly one parallel iteration (with finite-alphabet messages). Every node operates in unison, processing any particular local measurement in just two (discrete) decision stages: the first selects the symbols (if any) transmitted to its immediate neighbors and the second, upon receiving the symbols (or lack thereof) from the same neighbors, decides upon its local state estimate. We could just as well consider the case in which the neighbors communicating to any



(a) Undirected Topology (b) Equivalent Directed Topology

Figure 4.1. Illustration of the key step in our analysis of the simplest decision architecture with bidirectional inter-sensor communication: (a) an undirected network topology and (b) its "unraveled" directed counterpart, where each node is replicated as both a transmitter and a receiver.

particular node are different from those being communicated to by that node—for ease of exposition, we focus on the special case that these two types of neighbors are the same set of nodes. The formal mathematical model is described in Section 4.2.

Section 4.3 develops the team-theoretic analysis for this simplest undirected architecture. The key step is illustrated in Figure 4.1, where we "unravel" the bidirectional communication implied by an undirected topology into an equivalent directed topology in which each node appears both as a transmitter and a receiver. Though the resulting directed network is a polytree, because the node replication violates the critical conditional independence assumption, we cannot readily conclude that the tractable solution presented for directed networks in Chapter 3 is applicable. We prove it is applicable if the Bayesian cost function is separable across the nodes: specifically, under both the conditional independence and separable cost assumptions, the decision rules at every node reduce to a pair of local likelihood ratio tests. Moreover, the forward-backward offline algorithm defined on this equivalent directed topology translates into a parallel offline algorithm defined on the original undirected topology: in each offline iteration, every node exchanges both types of messages with all of its neighbors, firstly adjusting its stage-one rule and outgoing "likelihood" messages, then adjusting its stage-two rule and outgoing "cost-to-go" messages. This development is a positive result when contrasted with the simplest directed architecture considered in Chapter 3: the offline message-passing algorithm retains its correctness and convergence guarantees without restrictions on the (undirected) network topology.

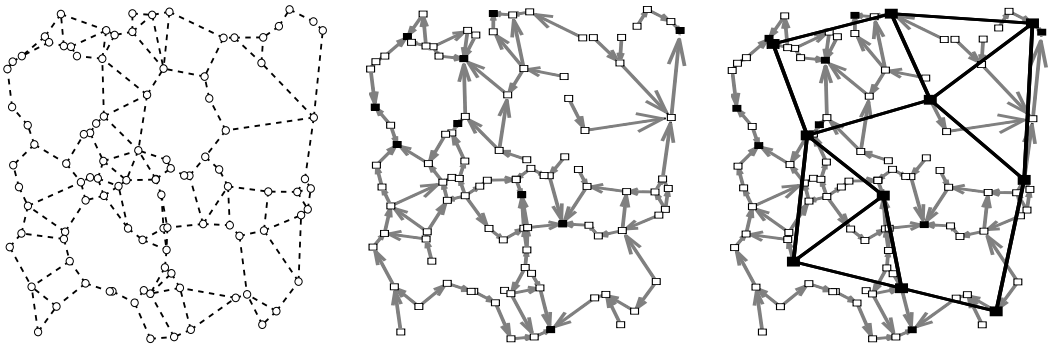
The basic idea of viewing bidirectional inter-sensor communication as a sequence of unidirectional decision stages has appeared in earlier research literature. A detailed analysis of two sensor nodes performing a global binary hypothesis test appears in [74]. Their model assumes one node is a primary decision-maker and the other acts as a

(costly) consultant, the latter only providing input when the former explicitly requests it. Indeed, their formulation satisfies the assumptions we require for tractability of the two-stage team solution in arbitrary n-node network topologies (and our analysis also accounts for the possibility of unreliable links). More general topologies or more than two decision stages (but still for a global binary hypothesis test and with reliable links) are considered in [2, 3, 36, 72, 103], but distinctly assuming that each node processes only a new measurement in every stage, essentially "forgetting" all of its preceding measurements and preserving the critical conditional independence assumption. In contrast, our problem formulation assumes each node processes the *same* local measurement over successive decision stages.[1] Though only a subtle difference in the online processing model, we show it gives rise to a new level of offline complexity: that is, the usual conditional independence assumption by itself does *not* imply that the optimal strategy admits a finite-dimensional parameterization.

With respect to a global decision objective of producing quality state estimates at every node, it is easily argued that allowing only a single online communication stage continues to over-constrain the problem. However, the impact of these constraints, in which *every* node is limited to online information within only its immediate neighborhood, is different from that of the one-sweep directed architecture considered in Chapter 3. In a directed network, nodes with more ancestors have advantage over those with few ancestors, while in an undirected network, nodes with more neighbors have advantage over nodes with few neighbors. The complementary aspects of these two different decision architectures motivate the consideration of hybrid network constraints to improve performance in problems for which neither type of network alone may be satisfactory. Section 4.4 considers a class of hybrid network constraints in which the online decision architecture is hierarchical, consisting of an (undirected) "leader" network atop of a (directed) "non-leader" network; see Figure 4.2. We show that combining the different offline message-passing algorithms in the natural way implied by the hybrid network constraints continues to satisfy team-optimality conditions and yields a convergent offline message-passing algorithm.

Section 4.5 closes this chapter with results from a number of experiments, using essentially the same class of local models used in the experiments of Chapter 3. The first series of experiments collectively illustrate that the choice between a directed or undirected architecture depends heavily on specific aspects of the problem, such as the

[1] Extension of our formulation and analysis to the case of multiple online communication stages is the subject of Chapter 5.



(a) Spatial Configuration (b) "Non-Leader" Network (c) Overlay "Leader" Network

Figure 4.2. An illustration the hybrid network constraints we analyze in Section 4.4. Starting from an arbitrary spatial configuration, as shown in (a), the "non-leader" network is any spanning directed acyclic subgraph (where the filled markers in (b) designate its childless nodes). The "leader" network is any undirected graph involving an arbitrary yet relatively small subset of the nodes in (a). Note that the leader network may connect nodes that are not necessarily spatial neighbors in (a), representing the (perhaps costly) opportunity for direct "long-distance" (e.g., multi-hop) online communication. Also note that the leader nodes in (c) need not necessarily coincide with the childless nodes in (b).

prior probabilities and Bayes costs as well as the particular directed and undirected network topologies being compared. Nonetheless, some general guidelines do emerge; for example, an undirected architecture is likely to be preferable when (i) many or all nodes are in the gateway and the network topology has small diameter in comparison to the number of nodes, and (ii) the hidden state processes are weakly-correlated. It is also often the case that global detection performance is best when the communication graph coincides with the probability graph, but we present some exceptions. Another set of experiments focuses on the presence of interference channels (i.e., the channel model of Example 3.2) but the absence of explicit communication-related costs (i.e., parameter $\lambda = 0$ in the multi-objective penalty function of (3.2)), clearly demonstrating how the offline message-passing algorithms account for the *implicit* informational costs of unreliable online communication. Our final set of experiments consider examples with hybrid network constraints, quantifying the performance gained by introducing a "leader" network.

■ 4.2 Online Processing Model

This section draws from the Bayesian decentralized formulation with costly and unreliable communication presented in Section 3.2, adapting it for the two-stage undirected

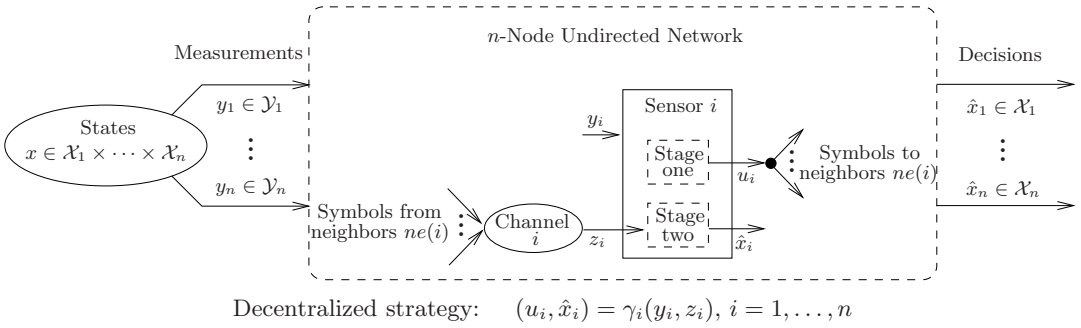


Figure 4.3. The n-sensor detection model described in Chapter 2, but assuming a decentralized decision strategy subject to network constraints defined on an n-node undirected graph, each edge representing a bidirectional (and perhaps unreliable) communication link between two spatially-distributed nodes. The online message schedule is constrained to exactly one parallel iteration in the network, every node processing its local measurement in just two decision stages: the first selects the symbols (if any) transmitted to its immediate neighbors and the second, upon receiving the symbols (or lack thereof) from the same neighbors, decides upon its local state estimate.

architecture depicted in Figure 4.3. The key difference from Chapter 3 is that the network topology $\mathcal{F}$ is undirected, where we assume every node i, initially observing only the component measurement y_i, operates in two distinct stages: the first stage decides upon the symbols $u_i \in \mathcal{U}_i$ (if any) transmitted to its neighbors[2] $ne(i) = \{j \mid \text{edge } (i,j) \text{ in } \mathcal{F}\}$ and the second stage, upon receiving the channel-corrupted symbols $z_i \in \mathcal{Z}_i$ from these same neighbors, decides upon the local estimate $\hat{x}_i \in \mathcal{X}_i$. Note that the rest of the model is essentially unchanged: we continue to assume (i) the hidden state x and observable measurement y take their values in, respectively, a discrete product space $\mathcal{X} = \mathcal{X}_1 \times \cdots \times \mathcal{X}_n$ and Euclidean product space $\mathcal{Y} = \mathcal{Y}_1 \times \cdots \times \mathcal{Y}_n$, (ii) each component of the global state estimate $\hat{x} \in \mathcal{X}$ is determined by an individual sensor and (iii) the collections of transmitted symbols u and received symbols z take their values in discrete product spaces $\mathcal{U} = \mathcal{U}_1 \times \cdots \times \mathcal{U}_n$ and $\mathcal{Z} = \mathcal{Z}_1 \times \cdots \times \mathcal{Z}_n$, respectively.

As before, the probabilistic model starts with a distribution $p(x, y)$ that jointly describes the hidden state process X and noisy measurement process Y. Given an undirected network topology $\mathcal{F}$, the decision processes $(U_i, \hat{X}_i)$ local to each node i are now generated sequentially: the stage-one decision rule defines the communication-related decision process U_i as a function of only the component measurement process Y_i, while the stage-two decision rule defines the detection-related decision process $\hat{X}_i$

[2]As discussed in Chapter 3, and illustrated in Examples 3.1–3.3, the symbol set $\mathcal{U}_i$ will reflect the particular transmission scheme employed by each node i.

as a function of both Y_i and U_i as well as the received information Z_i characterized by conditional distribution $p(z_i|x, y, u_{ne(i)})$ based on the information $U_{ne(i)} = \{U_j \mid j \in ne(i)\}$ collectively transmitted by the neighbors of node i. Let us denote by $\mathcal{M}_i$ all stage-one communication rules of the form $\mu_i : \mathcal{Y}_i \to \mathcal{U}_i$ and by Δ_i all stage-two detection rules of the form $\delta_i : \mathcal{Y}_i \times \mathcal{U}_i \times \mathcal{Z}_i \to \mathcal{X}_i$. Then, defining $\Gamma_i = \mathcal{M}_i \times \Delta_i$ for each node i, the admissible subset of decentralized strategies implied by $\mathcal{F}$ is given by $\Gamma = \Gamma_1 \times \cdots \times \Gamma_n$.

The decentralized design problem continues to be expressed by the multi-objective optimization problem in (3.2). However, the distribution that determines $J(\gamma)$ in (3.1) inherits a different structure as a result of the undirected network constraints. By the construction above, fixing the rules $\gamma_i = (\mu_i, \delta_i)$ local to node i is equivalent to specifying the distribution

$$p(u_i, \hat{x}_i|y_i, z_i; \gamma_i) = p(u_i|y_i; \mu_i)p(\hat{x}_i|y_i, u_i, z_i; \delta_i),$$

reflecting the two-stage causal processing implied by $\mathcal{F}$. It follows that fixing a strategy $\gamma \in \Gamma$ specifies the distribution

$$p(u, z, \hat{x}|x, y; \gamma) = \prod_{i=1}^{n} p\left(z_i|x, y, u_{ne(i)}\right) p(u_i, \hat{x}_i|y_i, z_i; \gamma_i)$$

and, in turn,

$$p(u, \hat{x}, x; \gamma) = \int_{y \in \mathcal{Y}} p(x, y) \prod_{i=1}^{n} p(u_i, \hat{x}_i|x, y, u_{ne(i)}; \gamma_i)\, dy, \tag{4.1}$$

where the summation over $\mathcal{Z}$ is taken inside the product i.e,

$$p(u_i, \hat{x}_i|x, y, u_{ne(i)}; \gamma_i) = \sum_{z_i \in \mathcal{Z}_i} p(z_i|x, y, u_{ne(i)}) p(u_i, \hat{x}_i|y_i, z_i; \gamma_i)$$

for each node i.

■ 4.3 Team-Theoretic Solution

This section summarizes the results of applying the team-theoretic analysis to the problem formulated in Section 4.2. As already depicted in Figure 4.1, the key idea is to map the set Γ of all two-stage strategies defined on an undirected topology $\mathcal{F}$ into an equivalent set of strategies defined on a particular two-level directed topology. In contrast to the results for the directed case in Chapter 3, our first result is a negative one: specifically, the usual conditional independence assumption does *not* by itself imply that the

team-optimal strategy admits a finite-dimensional parameterization. Our second result establishes that another assumption is needed, namely that the Bayesian cost function is separable across the nodes, for the optimal rules to take the form of likelihood-ratio tests (with measurement-independent thresholds). When both assumptions hold, the team optimality conditions reduce analytically to a nonlinear fixed-point equation with identical structure to that which arises for the "unraveled" directed counterpart. In turn, the forward-backward message-passing algorithm developed for directed polytrees immediately applies, translating into a parallel message-passing algorithm on the original undirected topology.

■ 4.3.1 Necessary Optimality Conditions

We begin the team-theoretic analysis for the design problem formulated in Section 4.2 by showing that the usual conditional independence assumption is not enough to guarantee that the optimal decentralized strategy γ^* in (3.2) admits a finite parameterization. Recall from Chapter 3 that, in the directed case, under this assumption the global minimizer γ^* in (3.2) reduces to a collection of likelihood-ratio tests, the parameters $\theta_i \in \mathbb{R}^{|\mathcal{U}_i \times \mathcal{X}_i \times \mathcal{X} \times \mathcal{Z}_i|}$ local to each node i coupled to the parameters $\theta_{-i} = \{\theta_j; j \neq i\}$ at all other nodes via the nonlinear fixed-point equation in (3.10). That parameter vector $\theta = (\theta_1, \ldots, \theta_n)$ is finite-dimensional is key to the correctness and convergence guarantees in Corollary 3.1.

Assumption 4.1 (Conditional Independence)**.** *For every node i,*

$$p(y_i, z_i | x, y_{\setminus i}, z_{\setminus i}, u_{\setminus i}) = p(y_i|x)p(z_i|x, u_{ne(i)}).$$

Proposition 4.1 (Person-by-Person Optimality)**.** *Let Assumption 4.1 hold. Consider any particular node i and assume both rules local to all other nodes are fixed at their optimal values in (3.2), which we denote by $\gamma^*_{\setminus i} = \{\gamma^*_j \in \Gamma_j \mid j \neq i\}$.*

• *Assume the stage-two rule local to node i is fixed at its optimal value $\delta^*_i \in \Delta_i$. The optimal stage-one rule reduces to*

$$\mu^*_i(Y_i) = \arg\min_{u_i \in \mathcal{U}_i} \sum_{x \in \mathcal{X}} a^*_i(u_i, x; Y_i) p(Y_i|x), \tag{4.2}$$

where the parameter values $a^*_i \in \mathbb{R}^{|\mathcal{U}_i \times \mathcal{X} \times \mathcal{Y}_i|}$ depend on all other fixed rules through a nonlinear operator f^1_i of the form

$$a^*_i = f^1_i(\delta^*_i, \gamma^*_{\setminus i}). \tag{4.3}$$

• *Assume the stage-one rule local to node i is fixed at its optimal value $\mu_i^* \in \mathcal{M}_i$. The optimal stage-two rule reduces to*

$$\delta_i^*(Y_i, U_i, Z_i) = \arg\min_{\hat{x}_i \in \mathcal{X}_i} \sum_{x \in \mathcal{X}} b_i^*(\hat{x}_i, x; U_i, Z_i) p(Y_i|x), \tag{4.4}$$

where parameter values $b_i^* \in \mathbb{R}^{|\mathcal{X}_i \times \mathcal{X} \times \mathcal{U}_i \times \mathcal{Z}_i|}$ depend upon all other fixed rules through a nonlinear operator f_i^2 of the form

$$b_i^* = f_i^2(\mu_i^*, \gamma_{\setminus i}^*). \tag{4.5}$$

Proof. Analogous steps as taken in the proof to Proposition 3.1; see Appendix B.1. □

It is instructive to contrast each part of Proposition 4.1 with Proposition 3.1 in the case of a directed network topology. The only difference in the stage-two rule δ_i^* is that the stage-one communication decision U_i acts as side information (in addition to local channel information Z_i); in particular, parameters b_i^* depend only on local measurement y_i through the discrete symbol $u_i = \mu_i^*(y_i)$, so the likelihood function $p(Y_i|x)$ is the sufficient statistic for process Y_i. However, in the stage-one rule μ_i^*, parameters a_i^* are seen to depend explicitly on the local measurement y_i. This structure is equivalent to that arising when Assumption 3.1 is violated for even the simplest directed networks (e.g., two nodes in series with discrete sets $\mathcal{Y}_1$ and $\mathcal{Y}_2$), in which case the decentralized design problem is known to be NP-complete [107]. Thus, Proposition 4.1 implies comparable complexity for the problem formulated in Section 4.2, which is a negative result compared to what is known for directed networks; that is, in contrast to the directed case, this (worst-case) complexity persists even when the conditional independence assumption holds. This negative result was largely anticipated in the earlier discussion of Figure 4.1, recognizing the equivalent directed network will comprise conditionally-*dependent* measurements.

From the algorithmic perspective, Proposition 4.1 tells us that the fixed-point equation of (3.10) still applies given the undirected model, with $\theta_i = (a_i, b_i)$ and $f_i = (f_i^1, f_i^2)$, but that the parameter vector θ need not necessarily be finite-dimensional. That is, the space of all finite collections of likelihood-ratio tests need not necessarily contain the optimal decentralized strategy γ^*. In essence, the fact that rule coefficients a_i^* depend explicitly on the local measurement $Y_i = y_i$ blurs the distinction between online and offline computation, and severs the associated equivalence between person-by-person optimality and solving a (finite-dimensional) nonlinear fixed-point equation. We now introduce an additional assumption and prove that it simultaneously alleviates the negative result and leads to a positive result: namely, the convergent offline algorithm

admits an efficient message-passing interpretation without restrictions on the (undirected) network topology (i.e., in contrast to the directed case, graph $\mathcal{F}$ need not be a tree).

Assumption 4.2 (Separable Costs)**.** *The global cost function in both stages of the decision process is additive over nodes of the network,*

$$c(u, \hat{x}, x) = \sum_{i=1}^{n} \left[c(\hat{x}_i, x) + \lambda c(u_i, x) \right]. \tag{4.6}$$

We will need a piece of new notation: for each node i in undirected network $\mathcal{F}$, define its two-step neighborhood by $ne^2(i) = \bigcup_{j \in ne(i)} ne(j) - i$, which includes all of its immediate neighbors together with each such neighbor's neighbors other than itself (i.e., all nodes within distance two from node i). It turns out that each node's communication rule is coupled to those of its two-step neighborhood, resulting from the facts that (i) each node's detection rule incorporates information based on transmissions from all of its neighbors and (ii) any two nodes with a common neighbor can be at most a distance two apart.

Proposition 4.2 (Tractable Person-by-Person Optimality)**.** *If Assumption 4.2 holds, then Proposition 4.1 applies with (4.3) and (4.5) specialized to the proportionalities*

$$a_i^*(u_i, x; y_i) \propto \alpha_i^*(u_i, x) = p(x) \left[\lambda c(u_i, x) + C_i^*(u_i, x) \right]$$

and

$$b_i^*(\hat{x}_i, x; u_i, z_i) \propto \beta_i^*(\hat{x}_i, x; z_i) = p(x) P_i^*(z_i|x) c(\hat{x}_i, x),$$

respectively, where (i) the likelihood function $P_i(z_i|x)$ for received information Z_i depends upon the fixed stage-one rules in the immediate neighborhood $ne(i)$ through a nonlinear operator g_i of the form

$$P_i^*(z_i|x) = g_i(\mu_{ne(i)}^*)$$

and (ii) the cost-to-go function $C_i(u_i, x)$ for transmitted information U_i depends upon the fixed stage-two rules in the immediate neighborhood as well as the fixed stage-one rules in the two-step neighborhood $ne^2(i)$ through a nonlinear operator h_i of the form

$$C_i^*(u_i, x) = h_i(\mu_{ne^2(i)}^*, \delta_{ne(i)}^*).$$

Proof. We provide only a sketch here; see Appendix B.2 for full details. Starting from the proof to Proposition 4.1, the key step is to establish that the optimal local stage-two rule δ_i^* (assuming all other rules fixed) lies in the subset of Δ_i consisting of all functions of the form $\delta_i : \mathcal{Y}_i \times \mathcal{Z}_i \to \mathcal{X}_i$ and, in turn, we may assume without loss of generality that $p(\hat{x}_i|y_i, u_i, z_i; \delta_i^*) = p(\hat{x}_i|y_i, z_i; \delta_i^*)$. Applying this reduction to the stage-two rules of all other nodes leads to local stage-one parameters a_i^* that do not depend on Y_i. □

■ 4.3.2 Message-Passing Interpretation

It is straightforward to verify that the equations in the proof of Proposition 4.2 are equivalent to the equations in Proposition 3.2 for the "unraveled" $2n$-node directed (polytree) network in which (parentless) nodes 1 to n employ the rules μ_1^* to μ_n^*, while (childless) nodes $n+1$ to $2n$ employ the rules δ_1^* to δ_n^*. Hence, the efficient message-passing interpretation presented in Chapter 3 for directed networks is readily applicable.

Assumption 4.3 (Measurement/Channel/Cost Locality)**.** *In addition to the conditions of Assumption 4.1 and Assumption 4.2, the measurement and channel models*[3] *as well as both stages of the cost function local to node i depend only on the local state process X_i i.e.,*

$$p(y_i, z_i|x, y_{-i}, z_{-i}, u_{\setminus i}) = p(y_i|x_i)p(z_i|x_i, u_{ne(i)})$$

and

$$c(u_i, \hat{x}_i, x) = c(\hat{x}_i, x_i) + \lambda c(u_i, x_i).$$

Corollary 4.1 (Online & Offline Efficiency)**.** *If Assumption 4.3 holds, then Proposition 4.2 reduces to*

$$\mu_i^*(Y_i) = \arg\min_{u_i \in \mathcal{U}_i} \sum_{x_i \in \mathcal{X}_i} \alpha_i^*(u_i, x_i) p(Y_i|x_i)$$

with stage-one rule parameters $\alpha_i^* \in \mathbb{R}^{|\mathcal{U}_i \times \mathcal{X}_i|}$ given by

$$\alpha_i^*(u_i, x_i) \propto p(x_i) \left[\lambda c(u_i, x_i) + \sum_{j \in ne(i)} C_{j \to i}^*(u_i, x_i) \right], \tag{4.7}$$

and

$$\delta_i^*(Y_i, Z_i) = \arg\min_{\hat{x}_i \in \mathcal{X}_i} \sum_{x_i \in \mathcal{X}_i} \beta_i^*(\hat{x}_i, x_i; Z_i) p(Y_i|x_i)$$

[3] Detecting a jammer is one application in which X_i might appear in the local channel model.

with stage-two rule parameters $\beta_i^* \in \mathbb{R}^{|\mathcal{X}_i \times \mathcal{X}_i \times \mathcal{Z}_i|}$ *given by*

$$\beta_i^*(\hat{x}_i, x_i; z_i) \propto c(\hat{x}_i, x_i) \sum_{x_{ne(i)}} p(x_i, x_{ne(i)}) \sum_{u_{ne(i)}} p(z_i|x_i, u_{ne(i)}) \prod_{j \in ne(i)} P_{j\to i}^*(u_j|x_j); \quad (4.8)$$

each node i *produces both a likelihood message for every neighbor* $j \in ne(i)$ *given by*

$$P_{i\to j}^*(u_i|x_i) = \int_{y_i} p(y_i|x_i) p(u_i|y_i; \mu_i^*) dy_i \quad (4.9)$$

and a cost-to-go message for each neighbor $j \in ne(i)$ *given by*

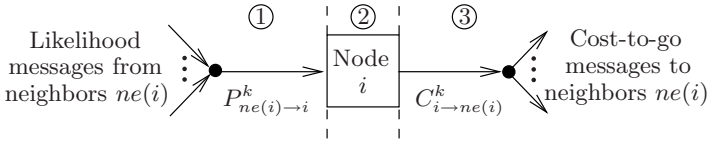
$$\begin{aligned} C_{i\to j}^*(u_j, x_j) = \sum_{x_i} \sum_{\hat{x}_i} c(\hat{x}_i, x_i) \sum_{x_{ne(i)-j}} p(x_i, x_{ne(i)}|x_j) \times \\ \sum_{u_{ne(i)-j}} p(\hat{x}_i|x_i, u_{ne(i)}; \delta_i^*) \prod_{m \in ne(i)-j} P_{m\to i}^*(u_m|x_m), \end{aligned} \quad (4.10)$$

$$p(\hat{x}_i|x_i, u_{ne(i)}; \delta_i^*) = \sum_{z_i} p(z_i|x_i, u_{ne(i)}) \int_{y_i} p(y_i|x_i) p(\hat{x}_i|y_i, z_i; \delta_i^*) dy_i.$$

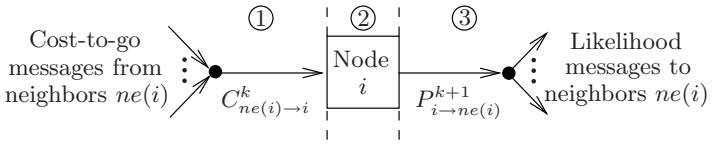
Proof. Corollary 3.2 and Proposition 3.2 starting from Proposition 4.2. □

Corollary 4.1 implies that the rule parameters $\phi_i^* = (\alpha_i^*, \beta_i^*)$ local to node i are completely determined by the incoming messages from neighbors $ne(i)$ on the original undirected network topology $\mathcal{F}$. Specifically, we see in (4.8) that the stage-two parameters β_i^* depend upon the incoming likelihood messages $P_{ne(i)\to i}^* = \{P_{j\to i}^*; j \in ne(i)\}$, the right-hand-side summarized by operator $f_i^2(P_{ne(i)\to i}^*)$. Meanwhile, we see in (4.7) that the stage-one parameters α_i^* depend upon the incoming cost-to-go messages $C_{ne(i)\to i}^* = \{C_{j\to i}^*; j \in ne(i)\}$, the right-hand-side summarized by operator $f_i^1(C_{ne(i)\to i}^*)$. Similarly, the satisfaction of Corollary 4.1 at all nodes other than i depends upon the outgoing messages from node i to its neighbors $ne(i)$. The outgoing likelihood messages $P_{i\to ne(i)}^* = \{P_{i\to j}^*; j \in ne(i)\}$ expressed in (4.9) are summarized by operator $g_i(\alpha_i^*)$, while the outgoing cost-to-go messages $C_{i\to ne(i)}^* = \{C_{i\to j}^*; j \in ne(i)\}$ expressed in (4.10) are summarized by operator $h_i(\beta_i^*, P_{ne(i)\to i}^*)$. Altogether, we see that Corollary 4.1 specializes the nonlinear fixed-point equations in (3.21) to

$$\begin{aligned} \alpha_i &= f_i^1\left(C_{ne(i)\to i}\right) \\ \beta_i &= f_i^2\left(P_{ne(i)\to i}\right) \\ P_{i\to ne(i)} &= g_i\left(\alpha_i\right) \\ C_{i\to ne(i)} &= h_i\left(\beta_i, P_{ne(i)\to i}\right) \end{aligned} \quad , \qquad i = 1, \ldots, n. \quad (4.11)$$



(a) Likelihood Step for Stage-Two Rule at Node i



(b) Cost-To-Go Step for Stage-One Rule at Node i

Figure 4.4. The kth parallel message-passing iteration as discussed in Corollary 4.2, each node i interleaving its purely-local computations with only nearest-neighbor communications.

Corollary 4.2 (Offline Message-Passing Algorithm)**.** *Initialize stage-one rule parameters $\alpha^0 = (\alpha^0_1, \ldots, \alpha^0_n)$ and stage-two rule parameters $\beta^0 = (\beta^0_1, \ldots, \beta^0_n)$, then generate the sequence $\{(\alpha^k, \beta^k); k = 1, 2, \ldots\}$ by iterating (4.11) in a parallel message schedule defined on the undirected graph $\mathcal{F}$, each node interleaving local updates of stage-one and stage-two decision rules with nearest-neighbor exchanges of likelihood and cost-to-go messages e.g., as illustrated in Figure 4.4,*

$$\begin{aligned}
P^k_{i\to ne(i)} &:= g_i\left(\alpha^{k-1}_i\right) && \text{from } i = 1, \ldots, n,\\
\beta^k_i &:= f^2_i\left(P^k_{ne(i)\to i}\right) && \text{from } i = 1, \ldots, n,\\
C^k_{i\to ne(i)} &:= h_i\left(\beta^k_i, P^k_{ne(i)\to i}\right) && \text{from } i = 1, \ldots, n \text{ and}\\
\alpha^k_i &:= f^1_i\left(C^k_{ne(i)\to i}\right) && \text{from } i = 1, \ldots, n.
\end{aligned}$$

If Assumption 4.3 holds, the associated sequence $\{J(\gamma^k)\}$ converges.

Proof. Corollary 3.3 starting from Corollary 4.1. □

Almost all of the remarks in Chapter 3 concerning the message-passing interpretation for directed networks carry over to the message-passing interpretation for undirected networks presented here. Firstly, it is *not* known whether the sequence $\{J(\gamma^k)\}$ converges to the optimal performance $J(\gamma^*)$, whether the achieved performance is invariant to the choice of initial parameters (α^0, β^0), nor whether the associated sequences $\{\alpha^k\}$ or $\{\beta^k\}$ converge. Secondly, each node need not possess a complete description of

the global state distribution $p(x)$ to carry out the message-passing iterations, as Corollary 4.1 implies it is sufficient for each node i to know the joint distribution $p(x_i, x_{ne(i)})$ of only the states local to itself and its neighbors. Thirdly, upon completion of the likelihood step in iteration $k+1$, computation of the global penalty $J(\gamma^k)$ scales linearly with n i.e.,

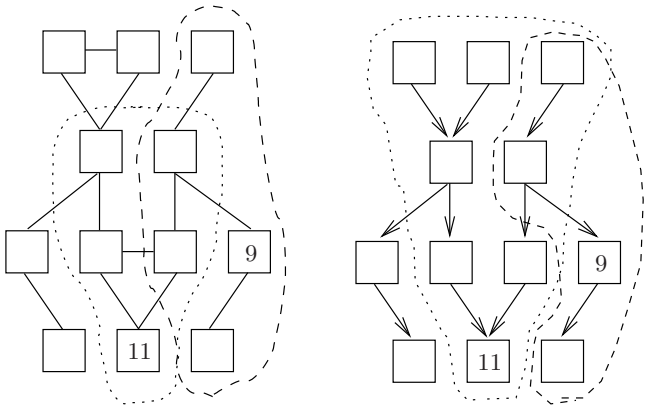
$$J(\gamma^k) := \sum_i \sum_{x_i} p(x_i) \left[\lambda G_i^1(\gamma^k|x_i) + G_i^2(\gamma^k|x_i) \right]$$

with

$$\begin{aligned} G_i^1(\gamma^k|x_i) &:= \sum_{u_i} c(u_i, x_i) p(u_i|x_i; \alpha_i^k), \\ G_i^2(\gamma^k|x_i) &:= \sum_{\hat{x}_i} c(\hat{x}_i, x_i) \sum_{z_i} p(\hat{x}_i|x_i, z_i; \beta_i^k) \sum_{u_{ne(i)}} p(z_i|x_i, u_{ne(i)}) \times \\ &\qquad \sum_{x_{ne(i)}} p(x_{ne(i)}|x_i) \prod_{j \in ne(i)} P_{j \to i}^{k+1}(u_j|x_j). \end{aligned}$$

An important difference from the case of directed networks is that the parallel message-passing algorithm in Corollary 4.2 retains its correctness and convergence guarantees without restrictions on the undirected topology (e.g., graph $\mathcal{F}$ need not be a tree). Also note that each type of network implies different explicit online constraints: in the directed case, each node's online data is related only to the measurements local to itself and its ancestors (i.e., its parents, the parents of each such parent, and so on); in the undirected case, each node's online data is related only to the measurements local to itself and its immediate neighbors. The different online processing constraints manifest themselves in different team couplings, in the sense discussed in Figure 4.5, being optimized by the respective offline message-passing algorithms. These architectural considerations suggest directed networks are preferable when comparably few nodes are to provide state estimates, while undirected networks are preferable when many nodes are to provide state estimates. In general, as will be demonstrated empirically in Section 4.5, such comparisons will also depend upon the particular topologies as well as the prior, measurement, channel or cost models.

We briefly mentioned in Section 4.1 that the online processing model in Section 4.2 (and, in turn, the results in this section) readily generalize to the possibility that neighbors communicating to a node, which we will call the node's *feeders*, are different from the neighbors being communicated to by that node, which we will call the node's *followers*. For example, consider an undirected network in which each node employs a selective peer-to-peer transmission scheme (as described in Example 3.3), yet there is



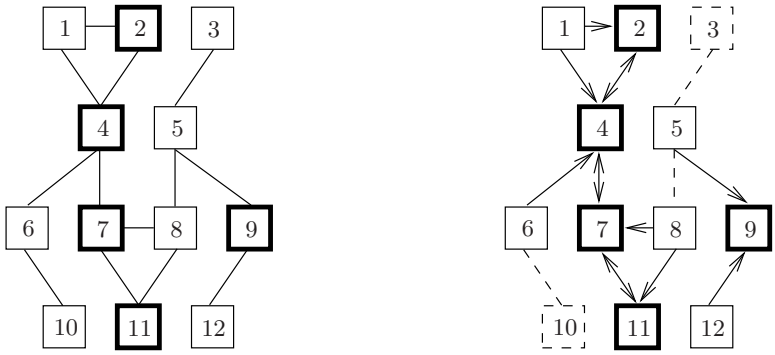
(a) Undirected Topology (b) Directed Topology

Figure 4.5. Comparison of the team coupling captured by the offline message-passing algorithm in (a) an undirected network or (b) a directed network. In (a), the incoming messages for each rule depend directly only on the rules of nodes within a two-step neighborhood (i.e., its immediate neighbors and the immediate neighbors of each such neighbor); in (b), the incoming messages depend directly upon the rules of all ancestors (a node's parents, the parents of each such parent, and so on), all descendants (i.e., a node's children, the children of each such child, and so on) as well as the ancestors of each such descendant. The dashed and dotted curves show these subsets for nodes 9 and 11, respectively, each of which similarly intersects with such subsets (not shown) of other nodes—the team coupling in each topology is the extent that the respective n subsets intersect.

at least one communication-only node (i.e., a node not in the gateway, meaning it need not make a local state-related decision). Clearly, within the single-stage undirected architecture, there is then no value in feeding information to any such communication-only node. Indeed, as examples in Section 4.5 will demonstrate, the offline message-passing algorithm converges to a strategy that shuts off (i.e., assigns zero use-rate to) every link entering a communication-only node. The point to be made here, however, is that we could equivalently have defined every node's followers to include only its neighboring gateway nodes; see Figure 4.6. The next section further exploits this inherent flexibility of the single-stage undirected architecture, allowing each node's followers to differ from its feeders and, in turn, broadening the class of detection networks for which our message-passing algorithms remain both efficient and convergent.

■ 4.4 Extension to Hybrid Network Constraints

Thus far, we have identified two simplest online decision architectures for which team-optimality conditions give rise to an offline iterative algorithm that admits an efficient message-passing interpretation. These are the single-sweep directed network of Chap-



(a) Undirected Network & Gateway Nodes (b) Directed Feeder/Follower Network

Figure 4.6. An (a) undirected network with only a strict subset of nodes in the gateway, shown by the thick-lined boxes, and (b) the equivalent directed feeder/follower network in which the directional arrows indicate information flow from a feeder to a follower (e.g., node 2 is a feeder and follower of node 4, but only a follower of node 1; meanwhile, node 1 is only a feeder of node 2 and node 4). The set of all followers are the gateway nodes (e.g., nodes 2,4,7,9 and 11), while the set of all feeders are all nodes except for those communication-only nodes having no neighboring gateway node (e.g., nodes 3 and 10).

ter 3 with Assumptions 3.2–3.4 in effect, and the single-iteration undirected network of the preceding subsections with Assumption 4.3 in effect. The complementary aspects of these two architectures, as was discussed by Figure 4.5, motivate the consideration of hybrid network constraints to improve detection performance in problems for which neither type of network alone may be satisfactory. This section identifies a special class of hybrid network constraints, along with assumptions under which negligible additional complexity is introduced in comparison to that identified for either type of network on its own.[4] Indeed, combining the efficient message-passing interpretations in the natural way implied by the hybrid network constraints is shown to retain analogous correctness and convergence guarantees to those of Corollary 3.3 and Corollary 4.2.

■ 4.4.1 Hierarchical Processing Model

We first introduce some notation associated with the class of hybrid network constraints illustrated in Figure 4.2. We are given a particular n-node directed acyclic graph $\mathcal{F}^D = (\mathcal{V}, \mathcal{E}^D)$ and a particular undirected graph $\mathcal{F}^U = (\mathcal{V}^U, \mathcal{E}^U)$ such that $\mathcal{V}^U \subset \mathcal{V}$, typically assuming $|\mathcal{V}^U|$ is much less than n. The former denotes the *non-leader* network and the latter the *leader* network, and the two together comprise an n-node *hybrid*

[4]More elaborate online decision architectures, in which there does arise additional complexity in comparison to the simplest architectures analyzed thus far, are the subject of Chapter 5.

network

$$\mathcal{H} = \mathcal{F}^D \cup \mathcal{F}^U = (\mathcal{V}, \mathcal{E}^D \cup \mathcal{E}^U).$$

For every non-leader node $i \in \mathcal{V}\backslash\mathcal{V}^U$, all of the usual terminology associated with its position in the directed network $\mathcal{F}^D$ continues to apply. Its rule space Γ_i is no different from before, consisting of all functions $\gamma_i : \mathcal{Y}_i \times \mathcal{Z}_i \to \mathcal{U}_i \times \mathcal{X}_i$ in which the local communication model dictates both how symbol set Z_i relates to the composite symbol set $\mathcal{U}_{pa(i)}$ of its parents and how the symbol set $\mathcal{U}_i$ relates to its children $ch(i)$.

For every leader node $\ell \in \mathcal{V}^U$, the terminology associated with its position in the undirected network $\mathcal{F}^U$ similarly continues to apply. However, its two-stage processing rule may now incorporate incoming symbols from its parents in $\mathcal{F}^D$, taking values in a set $\mathcal{Z}_\ell^D$, and generate outgoing symbols for its children in $\mathcal{F}^D$, taking values in a set $\mathcal{U}_\ell^D$. These symbol sets are to be contrasted with their counterparts $\mathcal{Z}_\ell^U$ and $\mathcal{U}_\ell^U$ within the undirected network $\mathcal{F}^U$. In particular, each leader node is taken to have two distinct channel models, $p(z_\ell^D|x, y, u_{pa(\ell)})$ describing information Z_ℓ^D received from its parents in $\mathcal{F}^D$, triggering the communication rule for the undirected network, and $p(z_\ell^U|x, y, u_{ne(\ell)}^U)$ describing information Z_ℓ^U received from its neighbors in $\mathcal{F}^U$, triggering the communication rule for the directed network. Altogether, we continue to assume that $\Gamma_\ell = \mathcal{M}_\ell \times \Delta_\ell$, but the single-stage rule spaces $\mathcal{M}_\ell$ and Δ_ℓ are augmented to consist of all functions

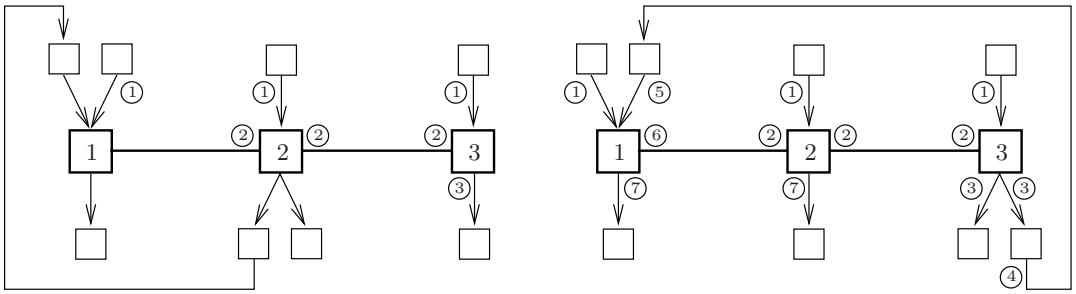
$$\mu_\ell : \mathcal{Y}_\ell \times \mathcal{Z}_\ell^D \to \mathcal{U}_\ell^U \qquad \text{and} \qquad \delta_\ell : \mathcal{Y}_\ell \times \mathcal{Z}_\ell^D \times \mathcal{U}_\ell^U \times \mathcal{Z}_\ell^U \to \mathcal{U}_\ell^D \times \mathcal{X}_\ell,$$

respectively. We will sometimes denote the product spaces $\mathcal{U}_\ell^D \times \mathcal{U}_\ell^U$ and $\mathcal{Z}_\ell^D \times \mathcal{Z}_\ell^U$ by $\mathcal{U}_\ell$ and $\mathcal{Z}_\ell$, respectively.

By the above construction, a leader node will not communicate within the undirected network (and, in turn, with any of its children in the directed network) until it has received symbols from all of its parents in the directed network. As illustrated in Figure 4.7, this opens up the possibility for *gridlock*, in which online processing can stall because information required before a leader node may begin cannot be realized until after this same leader node transmits information. The following assumption on hybrid network $\mathcal{H}$ ensures the absence of any such gridlock.

Assumption 4.4 (Absence of Gridlock)**.** *In hybrid network $\mathcal{H}$, for every pair of adjacent leader nodes in the undirected network $\mathcal{F}^U$, there exists no non-leader node that, in the directed network $\mathcal{F}^D$, is both an ancestor of one and a descendant of the other.*

With Assumption 4.4 in place, the strategy-dependent distribution that determines $J(\gamma)$ in (3.1) becomes well-defined. In particular, for each non-leader node $i \in \mathcal{V}\backslash\mathcal{V}^U$,



(a) Gridlock Between Leader Nodes 1 & 2 (b) Absence of Gridlock

Figure 4.7. Two simple hybrid network topologies, (a) one with gridlock and (b) one without gridlock. In each case, the leader (non-leader) nodes are the large (small) squares, while the circled numbers beside the links indicate the sequential partial-ordering of the nodes' communication decisions. By construction, each leader node is able to transmit to its neighboring leader nodes only after all of its non-leader parents have transmitted, and is similarly able to transmit to its non-leader children only after all of its neighboring leader nodes have transmitted. Note that (a) violates Assumption 4.4 while (b) does not.

fixing the rule $\gamma_i \in \Gamma_i$ is equivalent to specifying the distribution

$$p(u_i^D, \hat{x}_i | x, y, u_{pa(i)}^D; \gamma_i) = \sum_{z_i \in \mathcal{Z}_i} p(z_i | x, y, u_{pa(i)}^D) p(u_i^D, \hat{x}_i | y_i, z_i; \gamma_i).$$

Here, we have introduced the superscript-D notation on both u_i and $u_{pa(i)}$ for compatibility with the leader node notation, recognizing that (i) $u_i^D \equiv u_i \in \mathcal{U}_i$ for every non-leader node i and (ii) $u_j^D \equiv u_j \in \mathcal{U}_j$ for every parent $j \in pa(i)$ unless node j is also a leader node in which case $u_j^D \in \mathcal{U}_j^D$. For each leader node $\ell \in \mathcal{V}^U$, we use u_ℓ and z_ℓ to denote (u_ℓ^U, u_ℓ^D) and (z_ℓ^U, z_ℓ^D), respectively, so that similarly fixing the rule $\gamma_\ell = (\mu_\ell, \delta_\ell)$ is equivalent to specifying the distribution

$$p(u_\ell, \hat{x}_\ell | y_\ell, z_\ell; \gamma_\ell) = p(u_\ell^U | y_\ell, z_\ell^D; \mu_\ell) p(u_\ell^D, \hat{x}_\ell | y_\ell, z_\ell, u_\ell^U; \delta_\ell)$$

and, in turn,

$$p(u_\ell, \hat{x}_\ell | x, y, u_{pa(\ell)}^D, u_{ne(\ell)}^U; \gamma_\ell) =$$
$$\sum_{z_\ell^D \in \mathcal{Z}_\ell^D} p(z_\ell^D | x, y, u_{pa(\ell)}^D) \sum_{z_\ell^U \in \mathcal{Z}_\ell^U} p(z_\ell^U | x, y, u_{ne(\ell)}^U) p(u_\ell, \hat{x}_\ell | y_\ell, z_\ell; \gamma_\ell).$$

It follows that fixing the entire strategy $\gamma \in \Gamma_1 \times \cdots \times \Gamma_n$ specifies the distribution

$$p(u, \hat{x}, x; \gamma) = \int_{y \in \mathcal{Y}} p(x, y) \prod_{i \in \mathcal{V} \backslash V^U} p(u_i^D, \hat{x}_i | x, y, u_{pa(i)}^D; \gamma_i) \times \prod_{\ell \in \mathcal{V}^U} p(u_\ell, \hat{x}_\ell | x, y, u_{pa(\ell)}^D, u_{ne(\ell)}^U; \gamma_\ell)\, dy.$$

■ 4.4.2 Efficient Message-Passing Solutions

Given a hybrid network $\mathcal{H}$ that satisfies Assumption 4.4, team-optimality conditions and an associated message-passing interpretation follow from, essentially, a combination of the analytical steps taken in Chapter 3 and Section 4.3 when considering either type of network alone. Firstly, the optimal decentralized strategy γ^* is guaranteed to have a finite parameterization only under the conditional independence assumption and, at nodes also in the leader network, the assumption of separable costs. Secondly, in the case that X is itself a spatially-distributed random vector, we require the measurement/channel/cost locality assumption in order for total offline computation/communication overhead to scale only linearly with the number of nodes n. Thirdly, for the forward likelihood messages and backward cost-to-go messages to admit a recursive definition, we require the directed network $\mathcal{F}^D$ to be a polytree.

Assumption 4.5 (Conditional Independence & Measurement/Channel Locality)**.** *In hybrid network $\mathcal{H} = \mathcal{F}^D \cup \mathcal{F}^U$, the global probabilistic model satisfies*

$$p(y_i, z_i | x, y_{\backslash i}, z_{\backslash i}, u_{pa(i)}^D) = p(y_i | x_i) p(z_i | x_i, u_{pa(i)}^D)$$

for every non-leader node $i \in \mathcal{V} \backslash \mathcal{V}^U$ and

$$p(y_\ell, z_\ell | x, y_{\backslash \ell}, z_{\backslash \ell}, u_{pa(\ell)}^D, u_{ne(\ell)}^U) = p(y_\ell | x_\ell) p(z_\ell^D | x_\ell, u_{pa(i)}^D) p(z_\ell^U | x_\ell, u_{ne(\ell)}^U)$$

for every leader node $\ell \in \mathcal{V}^U$.

Assumption 4.6 (Separable Costs & Cost Locality)**.** *In hybrid network $\mathcal{H} = \mathcal{F}^D \cup \mathcal{F}^U$, the global cost function satisfies*

$$c(u, \hat{x}, x) = \sum_{i \in \mathcal{V}} c(u_i, \hat{x}_i, x_i)$$

with

$$c(u_i, \hat{x}_i, x_i) = \begin{cases} c(u_i^D, \hat{x}_i, x_i) & , \quad i \in \mathcal{V} \backslash \mathcal{V}^U \\ c(u_i^U, x_i) + c(u_i^D, \hat{x}_i, x_i) & , \quad i \in \mathcal{V}^U \end{cases}.$$

Assumption 4.7 (Polytree Non-Leader Network)**.** *In hybrid network $\mathcal{H} = \mathcal{F}^D \cup \mathcal{F}^U$, the directed (non-leader) network $\mathcal{F}^D$ is a polytree.*

The remainder of this section formally deduces that Assumptions 4.4–4.7 are *not* sufficient for the team-optimality conditions to admit an efficient message-passing interpretation. Additional restrictions on hybrid network $\mathcal{H} = \mathcal{F}^D \cup \mathcal{F}^U$, or more specifically on the interface between the directed (non-leader) network $\mathcal{F}^D$ and the undirected (leader) network $\mathcal{F}^U$, are required. These additional restrictions, subsuming Assumption 4.4 and Assumption 4.7, basically ensure that the "unraveled" hybrid network retains an overall directed polytree topology. Our approach considers two canonical types of hybrid network constraints, each illustrated in Figure 4.8: the first we call the hierarchical *fusion* architecture and the second we call the hierarchical *dissemination* architecture. In the former (latter), the directed non-leader network is said to feed (follow) the undirected leader network, analogous to the notions of "feeders" and "followers" we introduced in Figure 4.6 for purely undirected architectures. Our analysis proceeds by first establishing team-optimal message-passing equations for each canonical hybrid network, then combining these results to establish efficient message-passing equations for more general hybrid networks.

Proposition 4.3 (Hierarchical Fusion Architecture)**.** *In a hybrid network $\mathcal{H} = \mathcal{F}^D \cup \mathcal{F}^U$, let Assumptions 4.5–4.7 hold and suppose $\mathcal{F}^U$ is such that $\mathcal{V}^U = \{i \in \mathcal{V} | ch(i) = \emptyset\}$ i.e., the leader nodes are all childless nodes in $\mathcal{F}^D$ as shown in Figure 4.8(a). Unless there exists a non-leader node $i \in \mathcal{V} \backslash \mathcal{V}^U$ whose descendants (on $\mathcal{F}^D$) include a pair of leader nodes with distance between them less than or equal to two (on $\mathcal{F}^U$), the following message-passing equations satisfy team-optimality conditions.*

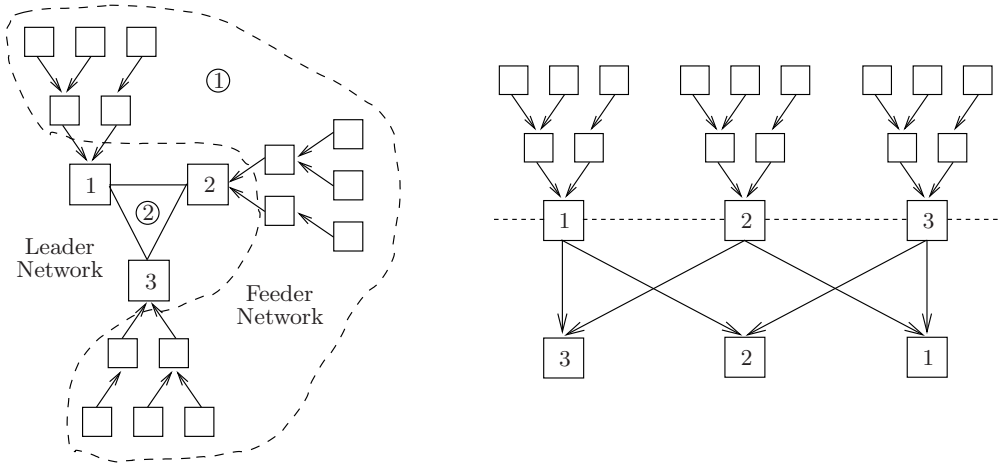
- *For every non-leader node $i \in \mathcal{V} \backslash \mathcal{V}^U$, rule parameters ϕ_i^*, forward messages $P^*_{i \to ch(i)}$ and backward messages $C^*_{i \to pa(i)}$ are as defined in Proposition 3.2.*

- *For every leader node $\ell \in \mathcal{V}^U$, the stage-one rule is given by*

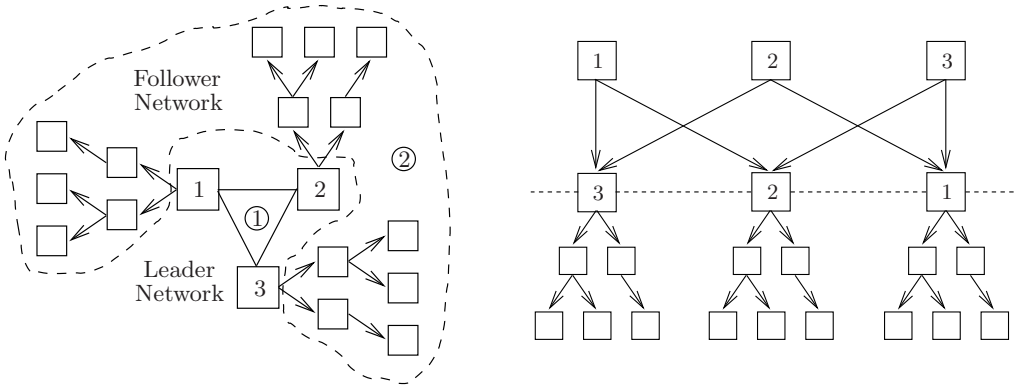
$$\mu_\ell^*(Y_\ell, Z_\ell^D) = \arg \min_{u_\ell^U \in \mathcal{U}_\ell^U} \sum_{x_\ell \in \mathcal{X}_\ell} \alpha_\ell^*(u_\ell^U, x_\ell; Z_\ell^D) p(Y_\ell | x_\ell),$$

where parameters $\alpha_\ell^* \in \mathbb{R}^{|\mathcal{U}_\ell^U \times \mathcal{X}_\ell \times \mathcal{Z}_\ell^D|}$ satisfy

$$\alpha_\ell^*(u_\ell^U, x_\ell; z_\ell^D) \propto p(x_\ell) P_\ell^*(z_\ell^D | x_\ell) \left[c(u_\ell^U, x_\ell) + \sum_{j \in ne(i)} C^*_{j \to \ell}(u_\ell^U, x_\ell) \right]$$



(a) A Hierarchical Fusion Architecture and Its Unraveled Counterpart



(b) A Hierarchical Dissemination Architecture and Its Unraveled Counterpart

Figure 4.8. Examples of two hierarchical decision architectures involving hybrid network constraints. The undirected network consists of three "leader" nodes, each connected to a distinct directed subtree of "non-leader" nodes. In (a), the flow of information begins with a single forward sweep in all non-leader networks and, upon every leader node hearing from its non-leader parents, ends with a single parallel iteration in the undirected network (i.e., the non-leader network is a feeder of the leader network). The opposite flow of information is assumed in (b), involving first the leader network and then the descendant non-leader networks (i.e., the non-leader network is a follower of the leader network).

with likelihood function $P^*_\ell(z^D_\ell|x_\ell)$ *defined analogously to (3.17) in Proposition 3.2; meanwhile, the stage-two rule, taking* $Z_\ell = (Z^D_\ell, Z^U_\ell)$*, is given by*

$$\delta^*_\ell(Y_\ell, Z_\ell) = \arg\min_{\hat{x}_\ell \in \mathcal{X}_\ell} \sum_{x_\ell \in \mathcal{X}_\ell} \beta^*_\ell(\hat{x}_\ell, x_\ell; Z_\ell) p(Y_\ell|x_\ell),$$

where parameters $\beta_\ell^* \in \mathbb{R}^{|\mathcal{X}_\ell \times \mathcal{X}_\ell \times \mathcal{Z}_\ell|}$ *satisfy*

$$\beta_\ell^*(\hat{x}_\ell, x_\ell; z_\ell) \propto p(x_\ell) P_\ell^*(z_\ell^D|x_\ell) P_\ell^*(z_\ell^U|x_\ell) c(\hat{x}_\ell, x_\ell)$$

with likelihood function

$$P_\ell^*(z_\ell^U|x_\ell) = \sum_{x_{ne(\ell)}} p(x_{ne(\ell)}|x_\ell) \sum_{u_{ne(\ell)}^U} p(z_\ell^U|x_\ell, u_{ne(\ell)}^U) \prod_{j \in ne(\ell)} P_{j\to\ell}^*(u_j^U|x_j); \tag{4.12}$$

each leader node ℓ *produces a likelihood message for every neighboring leader* $j \in ne(\ell)$ *given by*

$$P_{\ell\to j}^*(u_\ell^U|x_\ell) = \sum_{z_\ell^D} P_\ell^*(z_\ell^D|x_\ell) \int_{y_\ell} p(y_\ell|x_\ell) p(u_\ell^U|y_\ell, z_\ell^D; \mu_\ell^*) dy_\ell, \tag{4.13}$$

a cost-to-go message for each neighboring leader $j \in ne(i)$ *given by*

$$C_{\ell\to j}^*(u_j^U, x_j) = \sum_{x_\ell} \sum_{\hat{x}_\ell} c(\hat{x}_\ell, x_\ell) \sum_{x_{ne(\ell)\setminus j}} p(x_\ell, x_{ne(\ell)}|x_j) \times \sum_{u_{ne(\ell)\setminus j}^U} p(\hat{x}_\ell|x_\ell, u_{ne(\ell)}^U; \delta_\ell^*) \prod_{m \in ne(\ell)\setminus j} P_{m\to\ell}^*(u_m|x_m),$$

$$p(\hat{x}_\ell|x_\ell, u_{ne(\ell)}^U; \delta_\ell^*) = \sum_{z_\ell} P_\ell^*(z_\ell^D|x_\ell) p(z_\ell^U|x_\ell, u_{ne(\ell)}^U) \int_{y_\ell} p(y_\ell|x_\ell) p(\hat{x}_\ell|y_\ell, z_\ell; \delta_\ell^*) dy_\ell,$$

as well as a cost-to-go message $C_{\ell\to j}^*(u_j^D, x_j)$ *for each neighboring non-leader* $j \in pa(\ell)$ *defined analogously to (3.20) in Proposition 3.2 based on the stage-one rule* μ_ℓ^*.

Proof. First observe that, because no leader node has any descendants, $\mathcal{H}$ trivially satisfies Assumption 4.4. All assumptions under which Proposition 3.2 applies are satisfied for every non-leader node. It suffices to show that, despite the presence of incoming information Z_ℓ^D at every leader node ℓ, Corollary 4.1 continues to apply to the leader network $\mathcal{F}^U$. All of the steps in the proofs are seen to carry through provided that the likelihood function associated with $Z_\ell = (Z_\ell^D, Z_\ell^U)$ and the information $U_{ne(\ell)}^U$ collectively transmitted by the neighboring leaders of node ℓ obeys the identity

$$p(z_\ell, u_{ne(\ell)}^U|x, z_{ne(\ell)}^D; \gamma) = p(z_\ell^D|x; \gamma) p(z_\ell^U|x, u_{ne(\ell)}^U) \prod_{j \in ne(\ell)} p(u_j^U|x, z_j^D; \gamma)$$

under every fixed strategy $\gamma \in \Gamma$. This condition is violated at leader node ℓ only if it shares an ancestor (in $\mathcal{F}^D$) with at least one of its neighboring leader nodes, in which case

$$p(z_\ell|x, z_{ne(\ell)}^D, u_{ne(\ell)}^U; \gamma) \neq p(z_\ell^D|x; \gamma) p(z_\ell^U|x, u_{ne(\ell)}^U)$$

or any two of its neighboring leader nodes share a common ancestor (again in $\mathcal{F}^D$), in which case

$$p(u^U_{ne(\ell)}|x, z^D_{ne(\ell)}; \gamma) \neq \prod_{j \in ne(\ell)} p(u^U_j|x, z^D_j; \gamma).$$

□

Proposition 4.4 (Hierarchical Dissemination Architecture)**.** *In a hybrid network $\mathcal{H} = \mathcal{F}^D \cup \mathcal{F}^U$, let Assumptions 4.5–4.7 hold and suppose $\mathcal{F}^U$ is such that $\mathcal{V}^U = \{i \in \mathcal{V}|pa(i) = \emptyset\}$ i.e., the leader nodes are all parentless nodes in $\mathcal{F}^D$ as shown in Figure 4.8(b). Unless there exists a non-leader node $i \in \mathcal{V}\backslash\mathcal{V}^U$ whose ancestors (on $\mathcal{F}^D$) include a pair of leader nodes with distance between them less than or equal to two (on $\mathcal{F}^U$), the following message-passing equations satisfy team-optimality conditions.*

• *For every non-leader node $i \in \mathcal{V}\backslash\mathcal{V}^U$, rule parameters ϕ^*_i, forward messages $P^*_{i\to ch(i)}$ and backward messages $C^*_{i\to pa(i)}$ are as defined in Proposition 3.2.*

• *For every leader node $\ell \in \mathcal{V}^U$, the stage-one rule μ^*_ℓ (and its parameters α^*_ℓ) as well as the forward messages $P^*_{\ell\to ne(i)}$ are as defined in Corollary 4.1; meanwhile, the stage-two rule is given by*

$$\delta^*_\ell(Y_\ell, Z_\ell) = \arg \min_{(u^d_\ell, \hat{x}_\ell) \in \mathcal{U}^D_\ell \times \mathcal{X}_\ell} \sum_{x_\ell \in \mathcal{X}_\ell} \beta^*_\ell(u^D_\ell, \hat{x}_\ell, x_\ell; Z_\ell) p(Y_\ell|x_\ell),$$

which is also equivalent to that of Corollary 4.1 except with an augmented decision space that includes the symbol(s) u^D_ℓ for its children $ch(\ell)$ and, accordingly, rule parameters β^*_ℓ and backward messages $C^*_{\ell\to ne(i)}$ are also equivalently defined except that in (4.8) and (4.10), respectively, each appearance of $c(\hat{x}_\ell, x_\ell)$ is replaced with

$$c(u^D_\ell, \hat{x}_\ell, x_\ell) + \sum_{j \in ch(\ell)} C^*_{j\to\ell}(u^D_\ell, x_\ell);$$

finally, the forward message to every child $j \in ch(\ell)$ is given by

$$P^*_{\ell\to j}(u^D_\ell|x_\ell) = \sum_{z^U_\ell} P^*_\ell(z^U_\ell|x_\ell) \sum_{\hat{x}_\ell} \int_{y_\ell} p(y_\ell|x_\ell) p(u^D_\ell, \hat{x}_\ell|y_\ell, z^U_\ell; \delta^*_\ell)\, dy_\ell$$

with likelihood function $P^*_\ell(z^U_\ell|x_\ell)$ given by (4.12).

Proof. First observe that, because no leader node has any ancestors, $\mathcal{H}$ trivially satisfies Assumption 4.4. All assumptions under which Proposition 3.2 applies are satisfied for every non-leader node. It suffices to show that, despite the presence of outgoing

information U_ℓ^D at every leader node ℓ, Corollary 4.1 continues to apply to the leader network $\mathcal{F}^U$. All of the steps in the proofs are seen to carry through provided that the cost-to-go function associated with the information U_ℓ^U transmitted to the neighboring leaders of node ℓ decomposes additively under every fixed strategy $\gamma \in \Gamma$. Now, if node ℓ shares a descendant (in $\mathcal{F}^D$) with at least one if its neighboring leaders or any two of its neighboring leaders share a descendant (again in $\mathcal{F}^D$), then it is no longer the case at this shared non-leader node i that

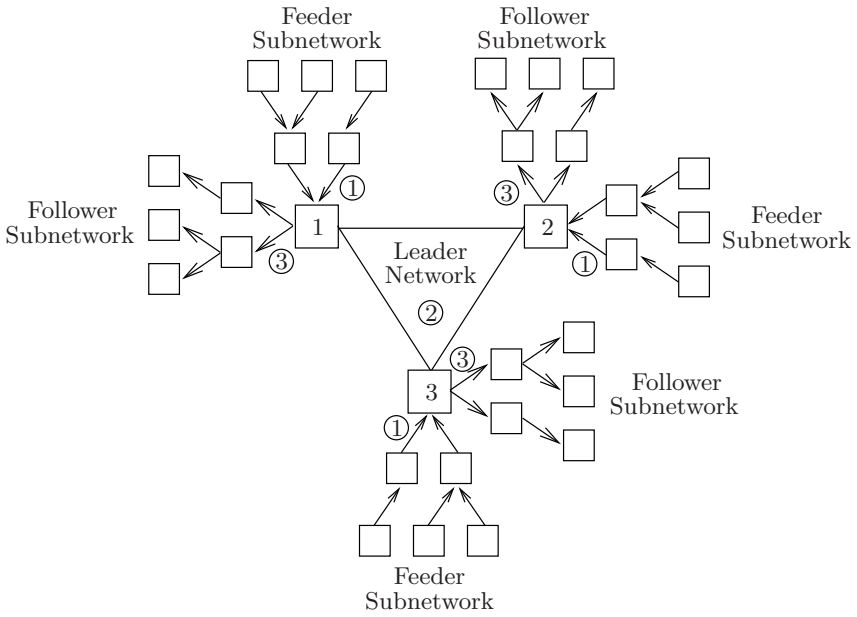
$$p(u_{pa(i)}|x, z_{pa(i)}; \gamma) = \prod_{j \in pa(i)} p(u_j|x, z_j; \gamma)$$

under every fixed $\gamma \in \Gamma$. In turn, the backward cost propagation from this node i need not necessarily decompose additively. □

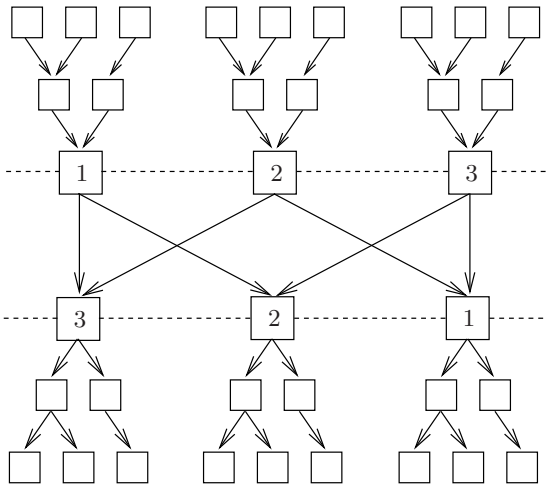
Combining Proposition 4.3 and Proposition 4.4 in the natural way, in which each leader node has both a feeder and follower network (e.g., see Figure 4.9), yields the general class of hybrid network constraints for which team-optimality conditions can reduce to efficient message-passing equations. To apply Proposition 4.3 to the leader network $\mathcal{F}^U$ and the feeder subnetworks, we require that, for every pair of leader nodes within a distance of two (on $\mathcal{F}^U$), the respective pairs of ancestors (on $\mathcal{F}^D$) are disjoint. The same is required of the respective pairs of descendants (on $\mathcal{F}^D$) to apply Proposition 4.4 to the leader network $\mathcal{F}^U$ and the follower subnetworks. The following assumption encapsulates the class of hybrid networks for which both Proposition 4.3 and Proposition 4.4 remain applicable, where the *lineage* of each leader node $\ell \in \mathcal{V}^U$ refers to the subset of nodes $an(\ell) \cup de(\ell)$, which may consist of both leader and non-leader nodes.

Assumption 4.8 (Hybrid Interface Restrictions)**.** *In a hybrid network $\mathcal{H} = \mathcal{F}^D \cup \mathcal{F}^U$, for every pair of leader nodes within a distance two of each other (on $\mathcal{F}^U$), the respective lineages (on $\mathcal{F}^D$) have no node in common: mathematically, for every $\ell \in \mathcal{V}^U$ and $m \in ne^2(\ell) \subset \mathcal{V}^U$, the intersection $(an(\ell) \cup de(\ell)) \cap (an(m) \cup de(m))$ is empty.*

Notice that the conditions in Assumption 4.8 subsume those of Assumption 4.4, as the latter is satisfied given every pair of adjacent leader nodes (i.e., leader nodes within a distance of one on $\mathcal{F}^U$) have disjoint lineages. The hybrid network in Figure 4.7(b), for example, satisfies Assumption 4.4 but violates Assumption 4.8. Intuitively-speaking, Assumption 4.8 (together with Assumption 4.7) ensures that the "unraveled" hybrid network retains an overall directed polytree topology. Specifically, the flow of online



(a) A hybrid network in which every leader node has both feeders and followers



(b) The "unraveled" directed counterpart of the hybrid network in (a)

Figure 4.9. A hybrid network formed from the natural junction of Figure 4.8(a) and Figure 4.8(b) at the leader network, allowing every leader node to have both feeder and follower subnetworks. This example lies in the class of hybrid networks for which team-optimality conditions reduce to efficient message-passing equations¿ The associated offline iterative algorithm admits a distributed implementation consisting of repeated forward-backward sweeps on the "unraveled" directed counterpart of (a).

measurement processing obeys the forward partial-order implied by this "unraveled" polytree, proceeding from parentless feeder nodes to childless follower nodes, where every leader node along the way, upon receiving symbols from all of its parents in $\mathcal{F}^D$, exchanges symbols with all of its neighbors in $\mathcal{F}^U$ before it transmits symbols to its children in $\mathcal{F}^D$. The offline message-passing algorithm similarly obeys this "unraveled" hybrid topology, making repeated forward-backward sweeps over the fixed-point equations obtained by combining Proposition 4.3 and Proposition 4.4.

Corollary 4.3 (Hybrid Offline Efficiency)**.** *Consider a hybrid network $\mathcal{H} = \mathcal{F}^D \cup \mathcal{F}^U$ and let Assumptions 4.5–4.8 hold. Team-optimality conditions are satisfied by the collection of non-leader rules*

$$\gamma_i^*(Y_i, Z_i) = \arg \min_{(u_i, \hat{x}_i) \in \mathcal{U}_i \times \mathcal{X}_i} \sum_{x_i \in \mathcal{X}_i} \phi_i^*(u_i, \hat{x}_i, x_i; Z_i) p(Y_i | x_i), \quad i \in \mathcal{V} \backslash \mathcal{V}^U$$

and the collection of leader rules

$$\begin{array}{rcl} \mu_\ell^*(Y_\ell, Z_\ell^D) & = & \arg \min\limits_{u_\ell^U \in \mathcal{U}_\ell^U} \sum\limits_{x_\ell \in \mathcal{X}_\ell} \alpha_\ell^*(u_\ell^U, x_\ell; Z_\ell^D) p(Y_\ell | x_\ell) \\ \delta_\ell^*(Y_\ell, Z_\ell) & = & \arg \min\limits_{(u_\ell^D, \hat{x}_\ell) \in \mathcal{U}_\ell^D \times \mathcal{X}_\ell} \sum\limits_{x_\ell \in \mathcal{X}_\ell} \beta_\ell^*(u_\ell^D, \hat{x}_\ell, x_\ell; Z_\ell) p(Y_\ell | x_\ell) \end{array}, \quad \ell \in \mathcal{V}^U$$

with real-valued parameters

$$\phi^* = \{\phi_i^*; i \in \mathcal{V} \backslash \mathcal{V}^U\} \bigcup \{(\alpha_\ell^*, \beta_\ell^*); \ell \in \mathcal{V}^U\}$$

denoting any solution to the following nonlinear fixed-point equations:

- *For every non-leader node $i \in \mathcal{V} \backslash \mathcal{V}^U$, we have*

$$\begin{array}{rcl} \phi_i & = & f_i \left(P_{pa(i) \to i}, C_{ch(i) \to i} \right) \\ P_{i \to ch(i)} & = & g_i \left(\phi_i, P_{pa(i) \to i} \right) \\ C_{i \to pa(i)} & = & h_i \left(\phi_i, P_{pa(i) \to i}, C_{ch(i) \to i} \right) \end{array}$$

with operators f_i, g_i and h_i based on equations described in Proposition 3.2.

- *For every leader node $\ell \in \mathcal{V}^U$, pertaining to the stage-one rule we have*

$$\begin{array}{rcl} \alpha_\ell & = & f_\ell^1 \left(P_{pa(\ell) \to \ell}, C_{ne(\ell) \to \ell} \right) \\ P_{\ell \to ne(\ell)} & = & g_\ell^U \left(\alpha_\ell, P_{pa(\ell) \to \ell} \right) \\ C_{\ell \to pa(\ell)} & = & h_\ell^D \left(\alpha_\ell, P_{pa(\ell) \to \ell}, C_{ne(\ell) \to \ell} \right) \end{array}$$

with operators f_ℓ^1, g_ℓ^U and h_ℓ^D based on equations described in Proposition 4.3; meanwhile, pertaining to the stage-two rule we have

$$\begin{aligned} \beta_\ell &= f_\ell^2\left(P_{pa(\ell)\to\ell}, P_{ne(\ell)\to\ell}, C_{ch(\ell)\to\ell}\right) \\ P_{\ell\to ch(\ell)} &= g_\ell^D\left(\beta_\ell, P_{pa(\ell)\to\ell}, P_{ne(\ell)\to}\right) \\ C_{\ell\to ne(\ell)} &= h_\ell^U\left(\beta_\ell, P_{pa(\ell)\to\ell}, P_{ne(\ell)\to\ell}, C_{ch(\ell)\to\ell}\right) \end{aligned}$$

with operators f_ℓ^2, g_ℓ^D and h_ℓ^U based on equations described in Proposition 4.4 but also accounting for the composite side information $Z_\ell = (Z_\ell^D, Z_\ell^U)$, each appearance of $P_\ell(z_\ell^U|x_\ell)$ replaced with the product $P_\ell^*(z_\ell^D|x_\ell)P_\ell^*(z_\ell^U|x_\ell)$.

Proof. First recognize that Assumption 4.7 and Assumption 4.8 together satisfy all conditions on $\mathcal{H}$ required by Proposition 4.3 and Proposition 4.4. Then, for every leader node ℓ in $\mathcal{F}^U$, we apply the message-passing equations of Proposition 4.3 to its ancestors in $\mathcal{F}^D$ and its local stage-one rule μ_ℓ^*; similarly, we apply Proposition 4.4 to its descendants in $\mathcal{F}^D$, and the combination of Proposition 4.3 and Proposition 4.4 to the stage-two rule δ_ℓ^*. □

Corollary 4.4 (Offline Message-Passing Algorithm)**.** *Initialize all rule parameters*

$$\phi^0 = \{\phi_i^0; i \in \mathcal{V}\backslash\mathcal{V}^U\} \bigcup \{(\alpha_\ell^0, \beta_\ell^0); \ell \in \mathcal{V}^U\},$$

then generate the sequence $\{\phi^k\}$ by iterating the fixed-point equations in Corollary 4.3 in repeated forward-backward passes on the "unraveled" directed counterpart to hybrid network $\mathcal{H}$ i.e., the kth forward pass on $\mathcal{H}$ proceeds from parentless feeder nodes to childless follower nodes, evaluating

$$P_{i\to ch(i)}^k := g_i\left(\phi_i^{k-1}, P_{pa(i)\to i}^k\right), \qquad i \in \mathcal{V}\backslash\mathcal{V}^U$$

for each non-leader node and

$$\begin{aligned} P_{\ell\to ne(\ell)}^k &:= g_\ell^U\left(\alpha_\ell^{k-1}, P_{pa(\ell)\to\ell}^k\right) \\ P_{\ell\to ch(\ell)}^k &:= g_\ell^D\left(\beta_\ell^{k-1}, P_{pa(\ell)\to\ell}^k, P_{ne(\ell)\to\ell}^k\right) \end{aligned}, \qquad \ell \in \mathcal{V}^U$$

for each leader node, while the kth backward pass on $\mathcal{H}$ proceeds from childless follower nodes to parentless feeder nodes, evaluating

$$\begin{aligned} \phi_i^k &:= f_i\left(P_{pa(i)\to i}^k, C_{ch(i)\to i}^k\right) \\ C_{i\to pa(i)}^k &:= h_i\left(\phi_i^k, P_{pa(i)\to i}^k, C_{ch(i)\to i}^k\right) \end{aligned}, \qquad i \in \mathcal{V}\backslash\mathcal{V}^U$$

for each non-leader node and

$$\begin{aligned} \beta_\ell^k &:= f_\ell^2\left(P^k_{pa(\ell)\to\ell}, P^k_{ne(\ell)\to\ell}, C^k_{ch(\ell)\to\ell}\right) \\ C^k_{\ell\to ne(\ell)} &:= h_\ell^U\left(\beta_\ell^k, P^k_{pa(\ell)\to\ell}, P^k_{ne(\ell)\to\ell}, C^k_{ch(\ell)\to\ell}\right) \\ \alpha_\ell^k &:= f_\ell^1\left(P^k_{pa(\ell)\to\ell}, C^k_{ne(\ell)\to\ell}\right) \\ C^k_{\ell\to pa(\ell)} &:= h_\ell^D\left(\alpha_\ell^k, P^k_{pa(\ell)\to\ell}, C^k_{ne(\ell)\to\ell}\right) \end{aligned}, \qquad \ell \in \mathcal{V}^U$$

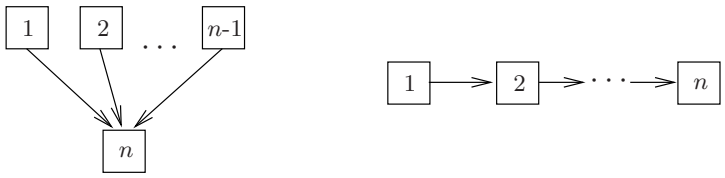
for each leader node. If Assumptions 4.5–4.8 hold, then the associated sequence $\{J(\gamma^k)\}$ *converges.*

Proof. Corollary 3.3 and Corollary 4.2 starting from Corollary 4.3. □

Inspection of the message-passing equations for hybrid networks shows that each non-leader node must know a local prior model related only to the directed network $\mathcal{F}^D$, while each leader node must know a local prior model related to both networks $\mathcal{F}^D$ and $\mathcal{F}^U$. Specifically, in order for leader node ℓ to exchange messages with its neighboring non-leaders in $\mathcal{F}^D$, it requires knowledge of $p(x_\ell, x_{pa(\ell)})$; similarly, in order to exchange messages with its neighboring leaders in $\mathcal{F}^U$, it requires knowledge of $p(x_\ell, x_{ne(\ell)})$. As was the case when we considered each architecture on its own, the scope of this thesis assumes these probabilities are available at initialization. It is also worth mentioning here that Assumption 4.8 in its full generality may be difficult to ensure in practice, requiring every node to acquire non-local properties of the overall hybrid topology. A more easily implemented special case (e.g., the example in Figure 4.9) is to further restrict the directed network $\mathcal{F}^D$ to be a forest, or a collection of disconnected polytrees, and form the undirected network $\mathcal{F}^U$ by choosing exactly one leader from each component polytree.

■ 4.5 Examples and Experiments

This section presents experiments with the offline message-passing algorithm for the undirected and hybrid architectures analyzed above. Throughout, the local measurement, channel and cost models are the same as those employed in the experiments of Chapter 3. Our primary purposes are threefold: firstly, we seek to compare the achievable detection performance when imposing single-iteration undirected constraints against that when imposing single-sweep directed constraints; secondly, we verify that our offline message-passing algorithms can capture not just explicit communication costs



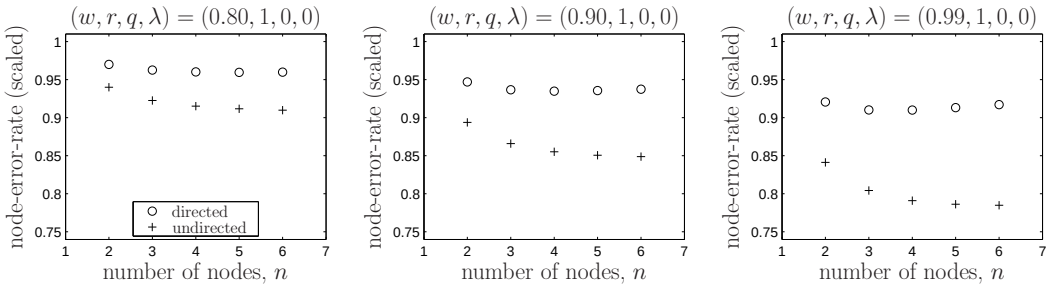
(a) Parallel (or Fusion) Topology (b) Series (or Tandem) Topology

Figure 4.10. The two most commonly studied online decision architectures in the decentralized detection literature. The experiments in Subsection 4.5.1 compare each of them to the single-iteration decision architecture implied by its undirected counterpart, holding all other models (i.e., the priors, likelihoods, channels, and costs) equal.

(as expressed in the penalty function J_c) but also *implicit*, or informational, costs for networks in which link erasures are not necessarily independent (as in the interference channel model of Example 3.2); and thirdly, we seek to quantify the gain in performance achieved by allowing hybrid architectures. Altogether, our architectural comparisons suggest the severity of performance differences depend heavily on other aspects of the problem, especially the degree of correlation between neighboring state processes and what subset of all nodes are in the gateway (i.e., what subset of nodes are responsible for producing local state-related decisions).

■ 4.5.1 Architectural Comparisons in Parallel & Series Topologies

In the decentralized detection literature, the two most commonly studied (directed) network topologies are the parallel (or fusion) topology and the series (or tandem) topology, both depicted in Figure 4.10. Many questions have been asked about how these two architectures compare when the global state process X is binary and the team objective is for node n to make the minimum-error-probability decision (and all finite-rate communication links are both reliable and cost-free). For example, in the case of just two nodes (technically $n = 3$ in Figure 4.10(a), but where the "fusion" node 3 receives no measurement of its own), the series topology is always better, as its admissible subset of strategies subsumes that of the parallel topology [76]. For a large number of (homogeneous) sensors, it is known that the parallel topology is always better, in the sense of an error exponent tending to zero as $n \to \infty$, while in the series topology this same error exponent is always bounded away from zero [75, 105]. Interestingly, the prediction (other than empirically e.g., [37]) of the largest number of sensors for which the series topology is still better than the parallel topology remains an open problem.



(a) Low State Correlation (b) Medium State Correlation (c) High State Correlation

Figure 4.11. Optimized node-error-rate performance in the *parallel* topology, under both directed and undirected network constraints, as a function of the number of nodes and for different degrees of state correlation. Each point is the node-error-rate achieved by the offline message-passing algorithm scaled by the node-error-rate of the myopic strategy. The undirected architecture is consistently favorable over the directed architecture; moreover, the (scaled) penalty of the undirected architecture is monotonically non-increasing as the number of nodes increases, which is untrue for the directed architecture.

The experiments in this subsection use our offline message-passing algorithms to (empirically) compare each directed architecture in Figure 4.10 with its undirected counterpart, assuming (i) that the state process X is a spatially-distributed random vector defined on a graphical model and (ii) the team objective is to minimize the expected number of nodes in error (and all nodes are in the gateway). In particular, we assume the same local models as in Subsection 3.4.1, fixing $\lambda = 0$ and $q = 0$ to represent cost-free communications over reliable (ternary-alphabet) links. We also assume a global prior $p(x)$ as defined in Subsection 3.4.2, parameterizing the state correlation by a common edge weight $w \in (0, 1)$, where in all cases the probability graph $\mathcal{G}$ is identical to the undirected counterpart of network topology $\mathcal{F}$. The point of these experiments is primarily to demonstrate that the message-passing algorithms perform as intuition would suggest: a secondary objective is to contrast the architectural issues for our model, namely n sensors performing n binary hypothesis tests with minimum sum-error rate, with the model of [75, 76, 105], namely n sensors performing a global binary hypothesis test with minimum error rate.

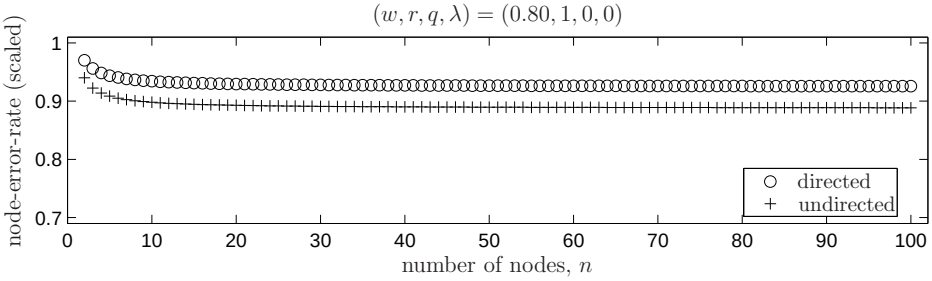
Figure 4.11 considers the parallel network topology of Figure 4.10(a) and compares the performance of the strategies obtained via our offline message-passing algorithms under the directed and undirected architectures. We see that the undirected architecture performs favorably relative to the directed architecture in all examples, with the largest difference being in the case of high state correlation and most sensors. This is easily explained by recognizing that node n receives comparable non-local information

in either type of architecture, but all other nodes receive non-local information only in the case of an undirected architecture. Alternatively, in the parallel topology, any directed strategy can always be viewed as a special case of an undirected strategy e.g., force node n to always sends the "0" symbol to nodes 1 to $n-1$. It is also worth noting that the (scaled) penalty of the undirected architecture is monotonically non-increasing as n increases, which we see is not necessarily true for the directed architecture. However, this monotonicity is observed for a parallel directed topology in [105]—in our model, adding a node provides a new measurement but also leads to more uncertainty in the global decision process, whereas in [105] adding a node simply provides a new measurement.

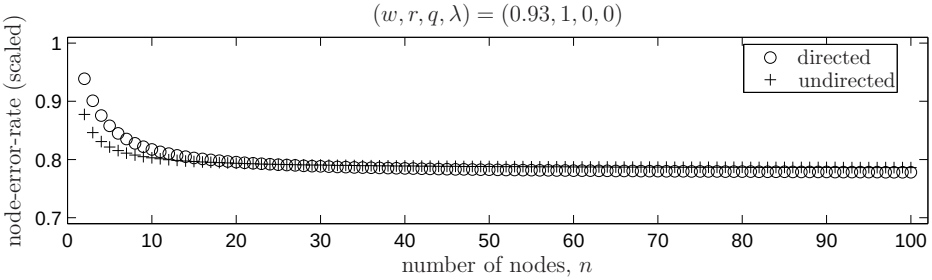
Figure 4.12 shows the analogous experimental results for the series topology of Figure 4.10(b). Note that with just two nodes, there is no distinction between the parallel and series topology and, as we expect from our discussion of Figure 4.11, the undirected architecture is always favorable. As the number of nodes in the series topology increases, however, an undirected architecture remains favorable only if the states are weakly correlated: in this case, the mixing time of the hidden process is comparable to the single-iteration reach of the undirected architecture, so the sequential yet unidirectional communication permitted by the directed architecture is of less value than the bidirectional communication permitted by the undirected architecture. However, as state correlation increases, the directed architecture permits well-informed decisions at the downstream sensors, and leads to favorable performance as the number of sensors grows. Not too surprisingly, the higher the state correlation, the fewer sensors are required before the directed architecture becomes favorable. The tandem model analyzed in [75] corresponds to ours in the case of extreme state correlation (i.e., edge-weight of $w = 1$), for which a directed architecture could well be favorable for every $n > 2$.

■ 4.5.2 Alternative Network Topologies

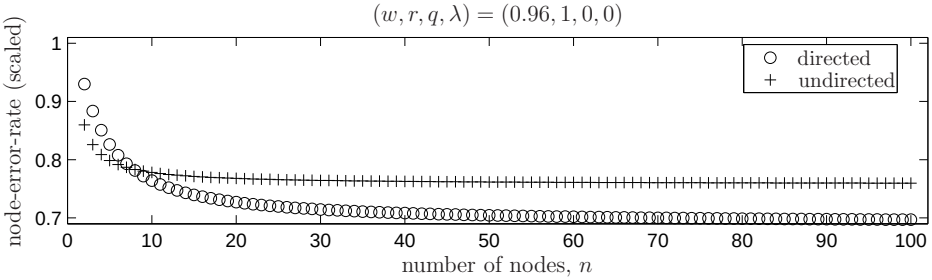
Recall from Chapter 2 that, given prior probabilities $p(x)$ defined by a tree structured graphical model, optimal centralized processing (i.e., computing posterior marginals at all nodes via belief propagation) requires communication only along the edges in the probability graph $\mathcal{G}$. For graphical models with cycles, while the belief propagation algorithm (assuming convergence) often provides good approximations to the posterior marginals, in the absence of convergence the approximation is poor, often performing worse than even the myopic solution. An intuitive idea for improvement is to allow message exchanges between non-neighboring nodes in the probability graph, which raises



(a) Low state correlation: undirected architecture favorable for all $n \leq 100$



(b) Medium state correlation: undirected architecture favorable for $n \leq 25$

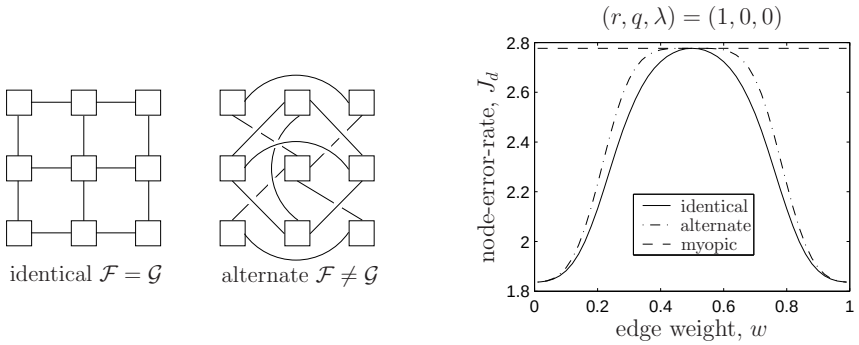


(c) High state correlation: undirected architecture favorable for $n \leq 7$

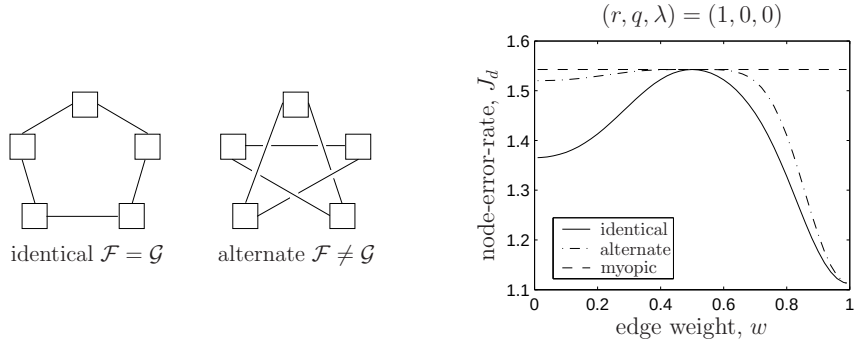
Figure 4.12. Optimized node-error-rate performance in the *series* topology, under both directed and undirected network constraints, as a function of the number of nodes and for different degrees of state correlation. In contrast to the results in Figure 4.11, whether the directed or undirected architecture is favorable depends upon both the number of nodes and the degree of state correlation.

a number of new questions: for example, which pairs of non-adjacent nodes do we choose, and how should these messages be both generated by the transmitting node and interpreted by the receiving node?

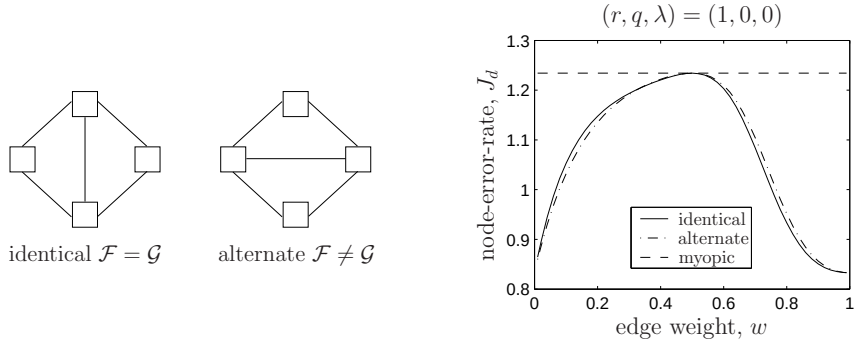
The results in Figure 4.13 summarize our experiments that consider the prospect of non-identical probability and communication graphs in simple "loopy" graphical mod-



(a) Comparison of Two Network Topologies in a 3-by-3 Nearest-Neighbor Grid Model

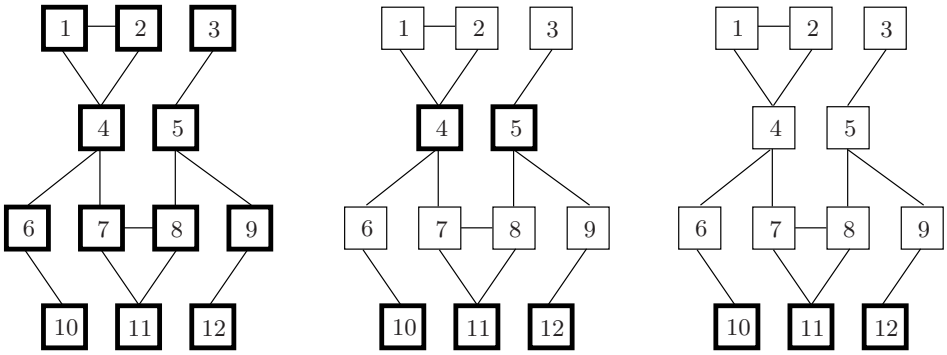


(b) Comparison of Two Network Topologies in a 5-Node Single-Cycle Model



(c) Comparison of Two Network Topologies in a 4-Node Triangulated Model

Figure 4.13. Performance comparison of identical and alternative undirected network topologies, all other things equal, for prior probabilities $p(x)$ defined by different "loopy" graphical models. Note that, in all three of the graphical models considered, the alternative network has the same number of edges as the identical network, and thus online communication overhead is also the same. Altogether, the results suggest that identical probability and communication graphs are typically preferable in our solution, but not always as demonstrated by (c) for "repulsive" edges (i.e. for edge weights $w < 0.5$).



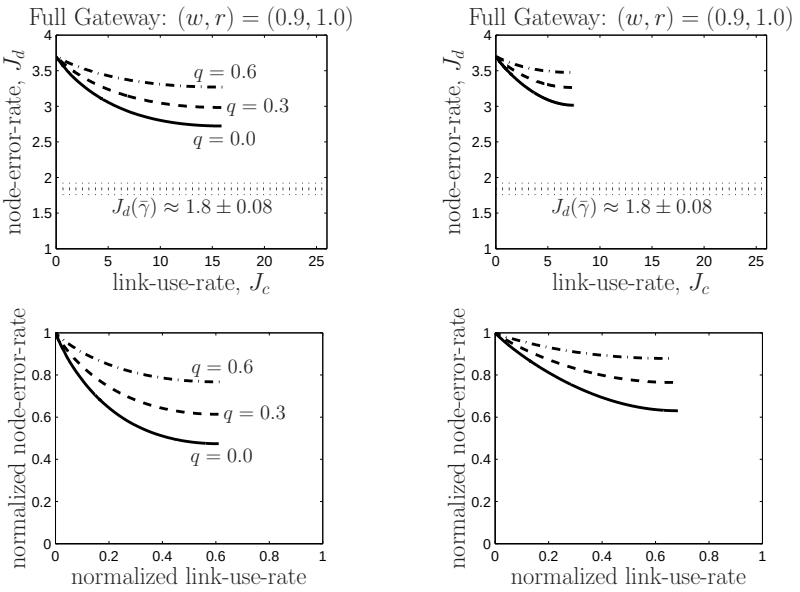
(a) Full Gateway (b) Half Gateway (c) Small Gateway

Figure 4.14. Three different gateways, indicated by the thick-lined markers, assumed in our empirical comparisons between an undirected architecture (with topology shown here) and a directed architecture (with topology shown in Figure 3.6(b)) for the twelve-node model first analyzed in Subsection 3.4.2.

els, focusing exclusively on designing the message-passing rules subject to the single-iteration undirected architecture. While not addressing the above questions in the context of belief propagation algorithms *per se*, given our solution constrains to one iteration of online communication and relies on an offline optimization step, the results do suggest the existence of models for which it is beneficial to allow non-identical graph structures. Specifically, in the 4-node triangulated model in Figure 4.13(c), allowing a communication graph that differs from the probability graph leads to improved global detection performance over that with identical graphs. Interestingly, this phenomenon is observed only for edge weights w leading to a so-called "frustrated" model, a case in which the belief propagation approximation is known to have difficulty. However, this phenomenon is not observed in the other loopy models we considered, suggesting that whether there is benefit to non-identical graph structures is not only a matter of the graphical model having cycles.

■ 4.5.3 A Small Illustrative Network: Revisited

In this subsection, we revisit the example considered in Subsection 3.4.2 and generate similar performance curves for the case of undirected network constraints. Throughout, the undirected topology (see Figure 4.14) is taken to be identical to the (loopy) graphical structure of the probabilistic model for X, with edge weight parameter fixed at $w = 0.9$ (i.e., neighboring binary states are positively correlated) and measurement noise parameter fixed at $r = 1$. We seek to compare the achieved performance to that of the (polytree) directed network already considered in Subsection 3.4.2. We draw this



q	λ^*	k^*
0.0	0.2760	3.3 ± 0.82
0.3	0.1935	2.9 ± 0.62
0.6	0.1107	2.8 ± 0.58

(a) Undirected Network Topology

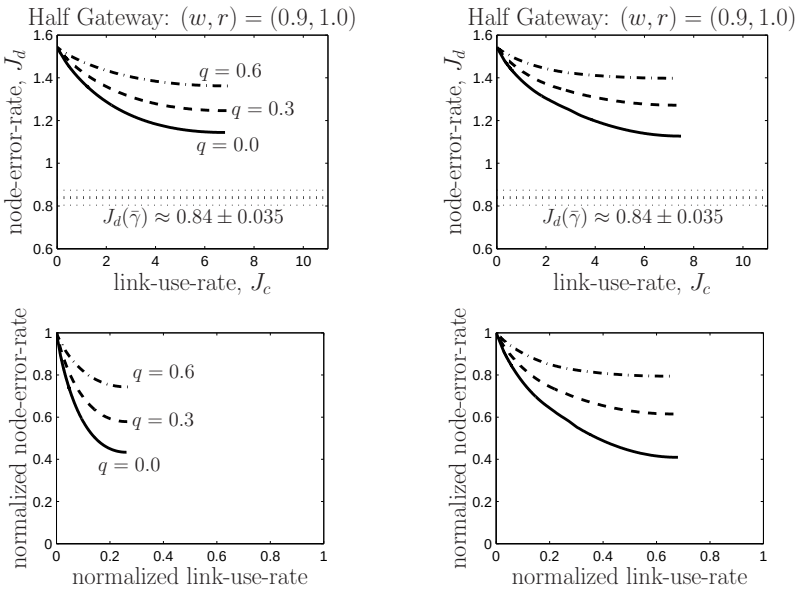
q	λ^*	k^*
0.0	0.2061	3.2 ± 0.44
0.3	0.1557	3.0 ± 0.17
0.6	0.0960	2.8 ± 0.42

(b) Directed Network Topology

Figure 4.15. Optimized tradeoff curves and tables for the full-gateway model discussed in Subsection 4.5.3 with parameter values $(w, r) = (0.9, 1)$ and $q \in \{0, 0.3, 0.6\}$, comparing results for (a) the undirected topology in Figure 4.14 and (b) the directed topology already analyzed in Subsection 3.4.2. Each curve is obtained by sampling λ in increments of 10^{-4}, starting with $\lambda = 0$, and declaring convergence in iteration k when $J(\gamma^{k-1}) - J(\gamma^k) < 10^{-3}$. Also shown is a Monte-Carlo estimate (plus or minus one standard deviation) of the centralized-optimal detection penalty $J_d(\bar{\gamma}^*)$, computed using 1000 samples from $p(x, y)$. The second row of figures uses the same data as the first, normalizing the penalties to better compare between the different topologies.

comparison for three different choices of gateway nodes, including the full gateway we assumed in Subsection 3.4.2, as well as a half-gateway consisting only of nodes $\{4, 5, 10, 11, 12\}$ and a small gateway consisting only of nodes $\{10, 11, 12\}$ as indicated in Figure 4.14.

Figures 4.15–4.17 display the tradeoffs between node-error-rate J_d and link-use-rate J_c achieved by the message-passing algorithms for the different network topologies and different gateways. In all cases, the same qualitative characteristics discussed in Subsection 3.4.2 for the directed topology are seen to carry over to the undirected topology.



q	λ^*	k^*
0.0	0.1767	3.4 ± 0.49
0.3	0.1308	3.0 ± 0.21
0.6	0.0789	3.0 ± 0.16

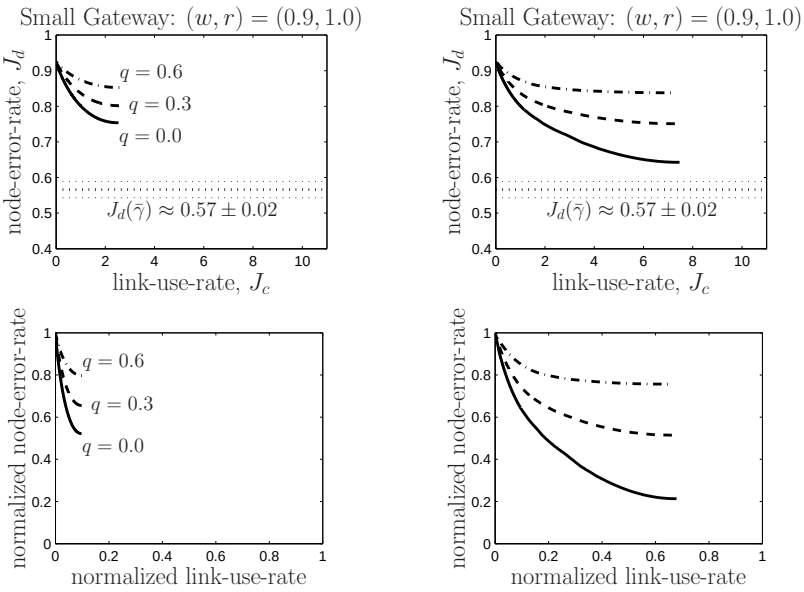
(a) Undirected Network Topology

q	λ^*	k^*
0.0	0.1686	3.1 ± 0.35
0.3	0.1317	3.0 ± 0.00
0.6	0.0837	3.0 ± 0.13

(b) Directed Network Topology

Figure 4.16. Analogous performance comparisons as presented in Figure 4.15, but assuming the half-gateway model in Figure 4.14. The performance curves turn out to be comparable on the absolute scale, implying a smaller normalized link-use-rate in the undirected topology, having 15 more unit-rate links than the directed topology. This stems from how, in the undirected architecture, the gateway nodes become "receive-only" nodes, learning offline that none of their neighbors benefit from whatever online information they could actively transmit. Similarly, the links between communication-only nodes 1 & 2 and 7 & 8, or the links that are absent in the directed architecture, also remain unused.

However, there are noteworthy quantitative differences. Firstly, for the full gateway, we see from Figure 4.15 that up to 55% of the optimal node-error-rate performance lost by the purely myopic strategy can be recovered in the undirected network, compared to 40% in the directed network. Of course, more link usage is inherent to the undirected topology (i.e., up to two bits per bidirectional edge, or 26 total bits per estimate, versus up to 11 total bits per estimate in the directed topology), but we see that the normalized link-use-rates are comparable. Figure 4.16 and Figure 4.17 illustrate the extent to which the undirected architecture results in an under-utilization of online communica-



q	λ^*	k^*
0.0	0.1767	3.0 ± 0.04
0.3	0.1308	3.0 ± 0.00
0.6	0.0789	3.0 ± 0.21

(a) Undirected Network Topology

q	λ^*	k^*
0.0	0.1686	3.0 ± 0.20
0.3	0.1317	3.0 ± 0.00
0.6	0.0837	2.8 ± 0.36

(b) Directed Network Topology

Figure 4.17. Analogous performance comparisons as presented in Figure 4.15, but assuming the small-gateway model in Figure 4.14. The single-iteration constraint of the undirected architecture is especially limiting for the small gateway, resulting in the extremely low link-use-rate and, ultimately, inferior detection performance relative to the directed architecture. Specifically, nodes 1 to 5 learn offline that they cannot contribute information to the gateway decisions and will thus sleep even in the special case of cost-free online communication (i.e., when λ approaches zero). Similarly, nodes 6 to 9 learn to selectively transmit only on the subset of links connected to the gateway nodes.

tion resources (and, ultimately, unsatisfactory detection performance in comparison to that achievable by the directed architecture) as fewer nodes are part of the gateway. These comparative trends in link usage are similarly reflected in the listed values of λ^* (i.e., our measure of the fair per-unit price of online communication), which for the full gateway are much larger in the undirected topology than in the directed topology relative to what is seen for the other two gateways. It is also seen that the listed values of k^* (i.e., our measure of the communication overhead for offline organization) for the two different topologies are comparable across all gateway scenarios.

These examples underscore that whether a directed and undirected architecture is

preferable depends strongly on the selected gateway nodes. It is also a simplest illustration of how the offline message-passing algorithms naturally capture the informational value of online communication: specifically, in the single-iteration undirected architecture, the algorithm always find a strategy where only the links into a gateway node get exercised, despite having been initialized otherwise, even in the absence of explicit communication penalty (i.e., when λ is zero); in the single-sweep directed architecture, as λ increases, nodes furthest from gateway nodes are the first to cease their active transmissions. A more compelling example of how the offline message-passing algorithm captures the informational value of online communication is considered next.

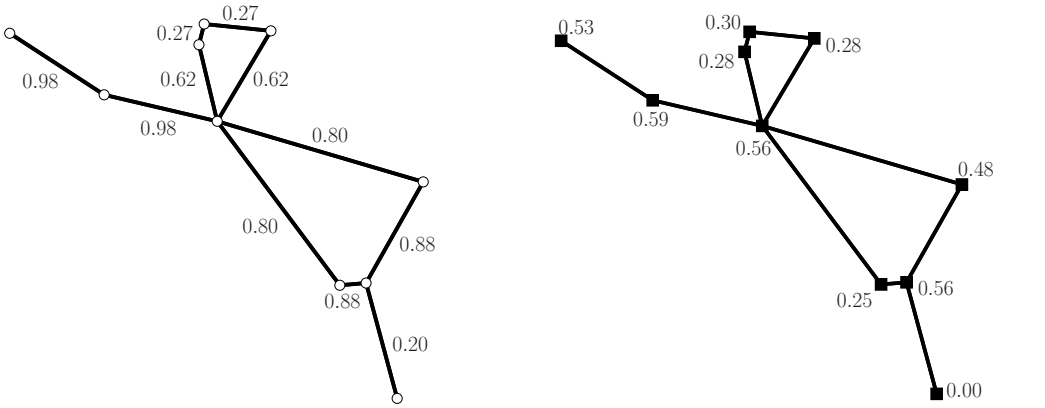
■ 4.5.4 Examples with Broadcast Communication and Interference

All examples considered thus far have used the peer-to-peer communication model with independent erasure channels as described in Example 3.1. This subsection focuses on examples that use the broadcast communication model with interference channels as described in Example 3.2, where nodes with two or more incoming links must contend with dependent erasures (i.e., whether any one symbol is erased depends on the value of the other symbol). That is, even if parameter q takes the same value at every node, the *effective* per-link erasure probabilities will be different, as they will depend on the network topology (i.e., in particular, on the degree of each node). Our results show that, qualitatively, the key tradeoffs and characteristics we've seen with independent erasure channels carry over to the case of interference channels; quantitatively, however, the value of online communication diminishes more rapidly as link reliability degrades (i.e., empirically, the value of λ^* decreases more rapidly as probability q increases).

The purpose of our first example is to illustrate how the strategies found by the offline message-passing algorithm can capture rather subtle issues arising from interference channels. Consider the ten-node undirected detection network depicted in Figure 4.18, having identical probability and communication graphs and assuming all nodes are in the gateway. The parameters of the local measurement models and (interference) channel models continue to be the same across all nodes, fixed at values $r = 1$ and $q = 0.2$, respectively. For the prior model, in contrast to all undirected graphical models considered thus far, we assign edge-dependent weights $w_{i,j} \in (0, 1)$ i.e., the edge potentials defined in Subsection 3.4.2 are generalized to the form

$$\psi_{i,j}(x_i, x_j) = \begin{cases} w_{i,j} & , \quad x_i = x_j \\ 1 - w_{i,j} & , \quad x_i \neq x_j \end{cases} .$$

The actual values given to these edge weights are not relevant to this discussion; what

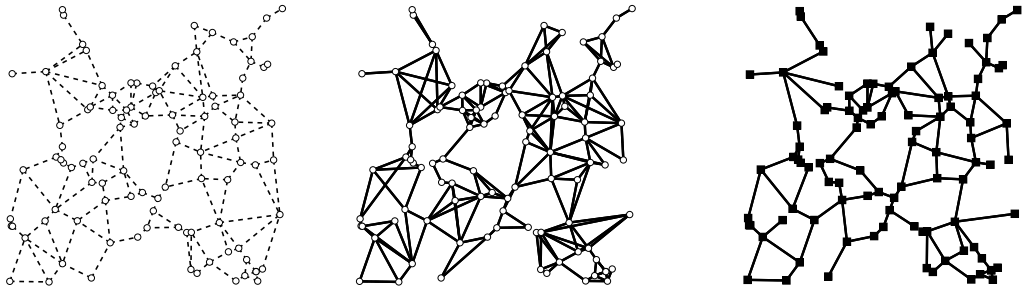


(a) Prior Probabilities: Pairwise Correlations (b) Optimized Strategy: Transmission Rates

Figure 4.18. A ten-node undirected detection network with identical probability and communication graphs, each node employing a selective broadcast scheme with interference parameter $q = 0.2$. The undirected graphical model yields the pairwise correlation coefficients (zero being uncorrelated and unity being always equal) shown in (a), where the bottom-most edge is associated to the weakest correlated pair of hidden states. The per-node transmission rates of the optimized strategy (assuming $r = 1$ for all nodes, all nodes are in the gateway and fixing λ to zero) are shown in (b), where the bottom-most node ceases to transmit at all. It has learned that, within the global detection objective, the implicit communication costs due to the interference channels outweighs the value of the information it can provide in support of its neighbor's final state-related decision.

is relevant is that they result in the pairwise correlation coefficients indicated in Figure 4.18(a). Specifically, notice that all adjacent state variables are positively correlated, with the least correlated pair of states associated to the bottom-most edge.

Figure 4.18(b) indicates the per-node broadcast transmission rates of the optimized strategy when $\lambda = 0$, in which case there are no explicit communication costs factored into the optimization. Nonetheless, we see that the bottom-most node has elected to be a "receive-only" node, though initialized otherwise before executing the offline message-passing algorithm. This is clear evidence that the algorithm can capture the implicit, or informational, costs resulting from the unreliable communication medium. That is, the bottom-most node has learned that transmitting to its neighbor would do more harm than good, interfering with the more informative transmissions coming from that neighbor's other neighbors. Recall that each node is initialized knowing nothing about the network topology beyond its neighborhood, nor about the local models used by nodes other than itself, so it is exclusively through the offline message-passing that the bottom-most node is able to arrive at this conclusion. The transmission rates exercised at other nodes follow a similar pattern, each node more likely to receive symbols from



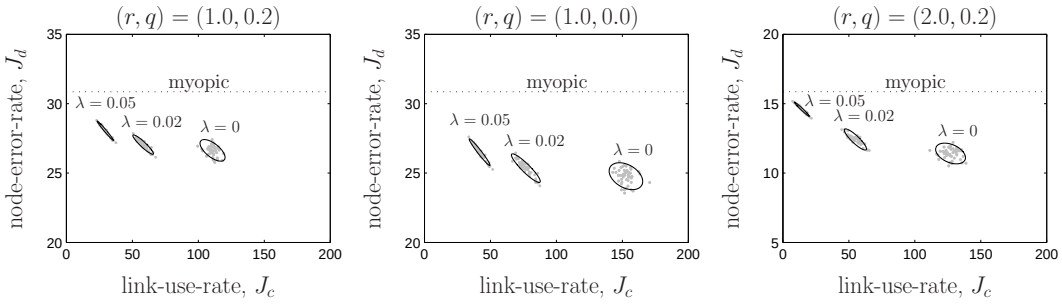
(a) Spatial Configuration (b) Undirected Graphical Model (c) Undirected Network Topology

Figure 4.19. A typical 100-node undirected detection network generated randomly for our experiments: (a) the spatial configuration of all nodes in the unit-area square, (b) the undirected graph $\mathcal{G}$ upon which the spatially-distributed state process X is defined and (c) the undirected network topology $\mathcal{F} \subset \mathcal{G}$, assuming all nodes are in the gateway.

those neighbors with the stronger pairwise correlations in Figure 4.18(a).

We now repeat the set of experiments discussed in Subsection 3.4.3 for randomly-generated 100-node detection networks, but considering undirected topologies and broadcast communication with interference (as opposed to the directed topologies and peer-to-peer communication with erasures). Also in contrast to the experiments in Subsection 3.4.3, here the gateway includes all 100 nodes. Figure 4.19 illustrates a typical output of our model generation procedure. The vector state process X is defined by the same directed graphical model described in Subsection 3.4.3, but we use the equivalent undirected graphical representation (as described in Chapter 2) to derive the undirected network topology $\mathcal{F}$. Specifically, the topology $\mathcal{F}$ is an arbitrary connected spanning subgraph of $\mathcal{G}$ in which each node is allowed to have at most five neighbors. Recall that local computation at each node scales exponentially in neighborhood size, so this restriction ensures that our randomly-generated model remains tractable.

Figure 4.20 depicts the average-case performance achieved by the parallel message-passing algorithm over 50 randomly-generated instances of an undirected detection network. Each plot consists of three clusters of 50 points, corresponding to the optimized performance for each model instance assuming three different values of λ. As we saw in Subsection 3.4.3 for directed topologies, we see here that our parallel message-passing algorithm, via parameter λ, consistently decreases global detection penalty (from that of the myopic strategy) as global communication penalty increases. Also shown for each optimized cluster is k^*, or the average number of iterations to convergence: interestingly, this price of our offline coordination appears to be much more consistent across all



λ	k^*
0.00	3.0 ± 0.00
0.02	4.0 ± 0.00
0.05	4.1 ± 0.35

(a) Nominal Environment

λ	k^*
0.00	3.0 ± 0.00
0.02	4.0 ± 0.14
0.05	4.0 ± 0.00

(b) Zero Channel Noise

λ	k^*
0.00	4.0 ± 0.00
0.02	4.3 ± 0.46
0.05	4.0 ± 0.00

(c) Low Measurement Noise

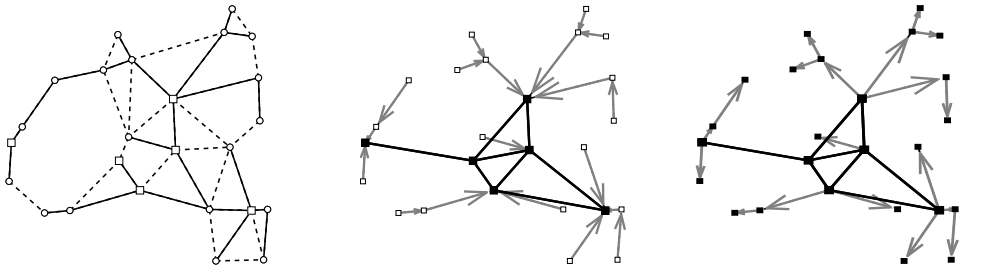
Figure 4.20. Performance of four different strategies for 50 randomly generated models of the type described in Subsection 4.5.4 given (a) a nominal environment, (b) zero channel noise and (c) low measurement noise. In each plot, the dotted horizontal line is the detection penalty achieved by the myopic strategy; the three clusters below this dotted line shows the performance of the optimized strategies for three different values of λ. Each ellipse is the least-squares fit to the 50 data points associated to each candidate strategy. In all cases, we declared convergence in iteration k when $J(\gamma^{k-1}) - J(\gamma^k) < 10^{-3}$, and each table lists the average number of offline iterations to convergence. See Subsection 4.5.4 for more discussion of these results.

scenarios than was the case for directed topologies.

■ 4.5.5 Benefits of Hybrid Network Constraints

In this subsection, we investigate the benefits of hybrid network constraints. This is done by comparing performance with and without the presence of the leader network, holding all other aspects of the problem constant. We will examine a randomly-generated 25-node hybrid network (see Figure 4.21), focusing on the two canonical cases discussed in Section 4.4, namely the hierarchical fusion (dissemination) architecture with the leader nodes consisting of all childless (parentless) nodes in the directed network $\mathcal{F}^D$. Our empirical results show performance benefits of using the (undirected) leader network $\mathcal{F}^U$ in both of these architectures, but with the greatest benefit in the hierarchical fusion architecture when the gateway nodes and leader nodes are one in the same.

In both sets of experiments, we assume a homogeneous network with measurement model parameter $r = 1$ and a selective broadcast communication scheme (see Example 3.3), obtaining results for three different values of interference probability



(a) Random Spatial Configuration (b) Fusion Architecture (c) Dissemination Architecture

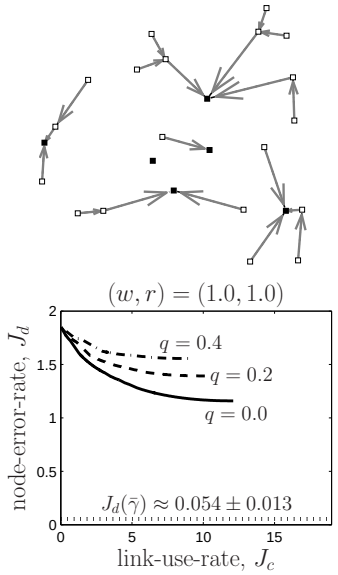
Figure 4.21. A (a) randomly generated 25-node spatial configuration, along with an embedded spanning tree (solid edges) and a randomly chosen subset of leader nodes (square markers) to initialize the construction of the hybrid forest topology, and the resulting hierarchical decision architectures used in the experiments described in Subsection 4.5.5. In both (b) and (c), the larger (smaller) markers denote the leader (non-leader) nodes and the filled (unfilled) markers denote the gateway (communication-only) nodes. In (b) the gateway and leader nodes are one in the same, while in (c) all of the nodes are in the gateway. Also note that the leader network includes edges between non-adjacent nodes in (a), while the non-leader network excludes edges in (a) as it seeks to ensure that Assumption 4.8 holds.

$q \in \{0.0, 0.2, 0.4\}$. For simplicity with respect to initializing each node with the requisite neighborhood priors, we assume the global state process X is a binary random variable (i.e., the trivial graphical model with edge weights $w = 1$, meaning $X_1 = X_2 = \cdots X_n$ with probability one) and equally-likely to takes its two possible values. That is,

$$p(x_i, x_{pa(i)}) = \begin{cases} 0.5 & , \text{ if } x_i = x_j \text{ for every } j \in pa(i) \\ 0 & , \text{ otherwise} \end{cases}$$

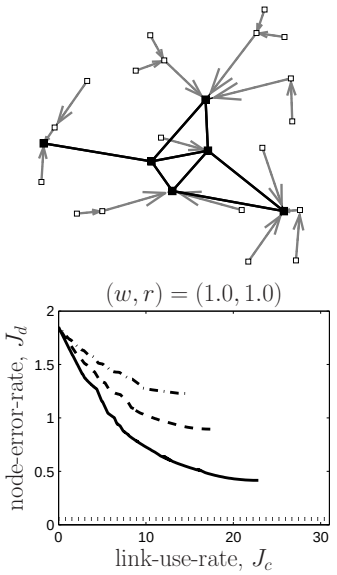
for every node i in $\mathcal{F}^D$, and the analogous expression for $p(x_i, x_{ne(i)})$ if node i is also in the leader network $\mathcal{F}^U$.

Figure 4.22 displays the optimized tradeoff curves for the hierarchical fusion architecture. Notice that our random construction of the hybrid network has left one leader node without any neighboring non-leaders. Nonetheless, with the leader network in place, up to roughly 80% of the detection performance lost by the myopic strategy (relative to the optimal centralized strategy) can be recovered; without the leader network, only up to roughly 40% of this performance gap is recovered. Of course, the leader network has a maximum of 35 active transmissions per global estimate (one transmission for each of the 19 directed links and two transmissions for each of the 8 undirected links) and, in turn, operates at higher total link use-rate (yet a comparable normalized link-use-rate). These different baseline link-use-rates brings any direct comparison of λ^*, or the fair per-unit price of online communication, between the two cases into question.



q	λ^*	k^*
0.0	0.2589	3.3 ± 0.75
0.2	0.2073	3.5 ± 0.99
0.4	0.1554	3.4 ± 1.2

(a) Without Leader Network

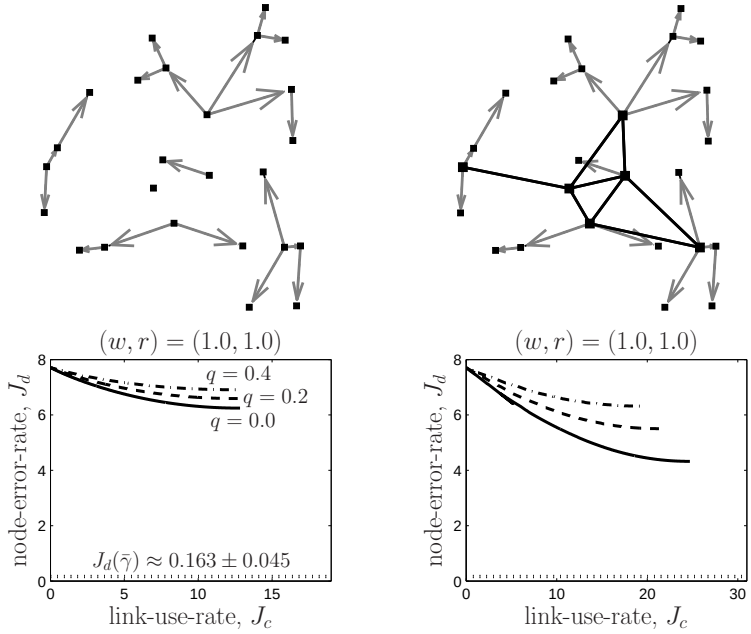


q	λ^*	$\frac{35}{19}\lambda^*$	k^*
0.0	0.1131	0.2083	5.7 ± 2.6
0.2	0.0807	0.1487	4.7 ± 1.2
0.4	0.0555	0.1022	4.3 ± 0.46

(b) With Leader Network

Figure 4.22. Comparison of the optimized tradeoff curves for the hierarchical fusion architecture in Figure 4.21(b) with and without the (undirected) leader network: in each case, we assume the interference channel model with $q = 0$ (solid line), 0.2 (dashed line) and 0.4 (dash-dotted line). Each curve is obtained by sampling λ in increments of 10^{-4}, starting with $\lambda = 0$, and declaring convergence in iteration k when $J(\gamma^{k-1}) - J(\gamma^k) < 10^{-3}$. Also shown is a Monte-Carlo estimate (plus or minus one standard deviation) of the centralized-optimal detection penalty $J_d(\bar{\gamma}^*)$, computed using 1000 samples from $p(x, y)$. The tables contain the two key quantities λ^* and k^* we record while computing each curve, respectively our empirical measures of the fair per-unit price of online communication resource and the resources consumed for coordination via the offline message-passing algorithm. See Subsection 4.5.5 for more discussion of these results.

As a zeroth-order approximation, we multiply the value of λ^* with the leader network in place by the fraction $\frac{35}{19}$, converting it to a per-unit price with the same number of links without the leader network in place. The resulting values also appear in the table of Figure 4.22b, and are consistently lower than the values of λ^* found without the leader network. In other words, all non-leader communication is significantly devalued (i.e., their links get used less frequently in the optimized strategy) upon introducing the leader network. This benefit has a price with respect to offline overhead, as k^* is



q	λ^*	k^*
0.0	0.2874	3.0 ± 0.38
0.2	0.2301	2.9 ± 0.33
0.4	0.1725	2.9 ± 0.28

(a) Without Leader Network

q	λ^*	k^*
0.0	0.2874	3.6 ± 0.84
0.2	0.2301	3.7 ± 0.82
0.4	0.1725	3.6 ± 0.83

(b) With Leader Network

Figure 4.23. Results for the same experiments as those associated with Figure 4.22, except considering the hierarchical dissemination architecture in Figure 4.21(c).

substantially larger with a leader network than without.

Figure 4.23 displays the optimized tradeoff curves for the hierarchical dissemination architecture. With the leader network in place, up to roughly 50% of the detection performance lost by the myopic strategy (relative to the optimal centralized strategy) can be recovered; without the leader network, only up to roughly 20% of this performance gap is recovered. In contrast to the hierarchical fusion architecture, we observe that λ^* has the same values whether or not the leader network is present. This reflects the fact that, as λ increases towards the critical value λ^*, the optimized hybrid network will always completely shut down the leader network before it completely shuts down the non-leader network. We again see an increased offline overhead as measured by k^* with the leader network in place than without, though this difference is seen to be less dramatic than was the case for the fusion architecture in Figure 4.22.

Chapter 5

On Multi-Stage Communication Architectures

IN the two preceding chapters, the principal focus was on decision architectures in which there is only a single stage of online communication. In the one-stage directed architecture of Chapter 3, every node receives information based only on its ancestors' measurements, and hence nodes at the beginning of the forward partial order face the greatest risk of making poor local state-related decisions; in the one-stage undirected architecture of Chapter 4, every node receives information based only on immediate neighbors' measurements, and hence nodes with smallest degree face the greatest risk of making poor local state-related decisions. While the hybrid architectures of Chapter 4 allow for "long-distance" information sharing between otherwise disconnected nodes, the underlying non-leader network continues to operate as a one-sweep directed architecture and, as such, nodes early in its forward partial order (and *not* part of the leader network) inherit a similar risk of making poor local state-related decisions.

■ 5.1 Chapter Overview

Looking towards applications in which all nodes are to make well-informed local state-related decisions, this chapter aims to generalize both the directed and undirected architectures of the preceding chapters to allow for multiple online communication stages. Section 5.2 formulates these multi-stage extensions mathematically, adopting message schedules analogous to those discussed in Chapter 2 for (optimal) belief propagation algorithms on tree-structured graphical models. Our model however assumes, as in our single-stage formulations, that the online network is constrained to low-rate or unreliable links and the associated communication graph need not be equivalent to the probability graph. Our approximation is also more goal-directed, in the sense that (on-

line) belief propagation seeks to map any particular global measurement $Y = y$ to the posterior marginals $p(x_i|y)$ for all i, while any candidate strategy in our formulation seeks to map $Y = y$ directly to the assigned values $\hat{x}_i$ for all i. Recall from Example 2.7 that the former is a sufficient statistic for the latter when costs are additive across the nodes e.g., when the objective is to minimize the expected number of nodes in error.

Section 5.3 applies the team-theoretic analysis of the previous chapters to these multi-stage formulations, exposing a number of new structural properties that an optimal decentralized strategy should satisfy. Specifically, we suggest how each local processing rule can make explicit use of memory, enabling each node to successively pare down its local likelihood in a most informative yet resourceful way as a function of *all* previously observed symbols (i.e., all symbols the node has both received and transmitted in previous stages). Unfortunately, even under best-case model assumptions (i.e., the analogous assumptions exploited in earlier chapters), the required memory (and, in turn, the offline solution complexity) grows exponentially with the number of communication stages. Interestingly, online computation continues to grow linearly with the number of nodes (given a sparsely-connected communication graph), and the expanding memory at every node essentially affords an increasingly-accurate approximation to the sufficient statistic (e.g., the node's posterior marginal) for making the local state-related decision.

The exposed barriers to tractably computing team-optimal decision strategies in multi-stage communication architectures motivate introducing additional approximations. Section 5.4 describes one such approximate offline algorithm, leveraging the efficient message-passing algorithms derived in previous chapters. The nature of this approximation makes it especially suitable to examine the extent to which performance improves when moving from one-stage to two-stage architectures, then from two-stage to three-stage architectures and so on. A number of small-scale experiments with the approximation are presented in Section 5.5, indicating that near-optimal decision performance is achievable in a number of stages comparable to the diameter of the network. These experiments also include using our multi-stage approximation to obtain estimates of all nodes' posterior marginals, making contact with ongoing research related to belief propagation (BP) [18, 47, 53, 66, 70, 77, 90, 116] and providing the first inroads into the issue of BP message quantization in the *low-rate* regime. Nevertheless, this chapter leaves many important theoretical and algorithmic questions about these multi-stage architectures unanswered, which will be discussed in Chapter 6 as opportunities for future research.

■ 5.2 Online Processing Models

This section generalizes the decentralized n-sensor detection problem expressed by the multi-objective optimization problem in (3.2) to the case of $T \geq 2$ communication stages. The joint distribution $p(x, y)$ for length-n random vectors X and Y, the given network topology $\mathcal{F}$ and the detection-related cost function $c(\hat{x}, x)$ are exactly as described in the preceding chapters. For every communication stage $t \leq T$, we let u_i^t denote the symbols (if any) transmitted by node i, taking values in a finite set $\mathcal{U}_i^t$. The actual cardinality of each set $\mathcal{U}_i^t$ will, exactly as was described for the single-stage architectures in the previous chapters, reflect the neighbors of node i in network topology $\mathcal{F}$ as well as the presumed transmission scheme (e.g., selective versus non-selective, peer-to-peer versus broadcast) local to node i. In any case, upon generalizing u_i for each node i to take values in the finite set $\mathcal{U}_i = \mathcal{U}_i^1 \times \cdots \times \mathcal{U}_i^T$, the communication-related cost function $c(u, x)$ may also be defined exactly as in previous chapters.

It is entirely through the admissible subset of strategies $\Gamma \subset \bar{\Gamma}$ that we will capture differences in multi-stage processing assumptions associated with the different types of network constraints. Given network topology $\mathcal{F}$ is undirected, the multi-stage architecture consists of T parallel communication stages, every node in each stage exchanging symbols with only its immediate neighbors. Given network topology $\mathcal{F}$ is directed, the multi-stage architecture consists of repeated forward-backward sweeps, each odd-numbered stage $t = 1, 3, \ldots$ communicating in the forward partial order and each even-numbered stage $t = 2, 4, \ldots$ communicating in the backward partial order. In any case, all local state-related decisions $\hat{x} = (\hat{x}_1, \ldots, \hat{x}_n)$ are made upon completion of the Tth communication stage. The following subsections more carefully describe the function space Γ of all multi-stage processing strategies for each type of network.

We will see that the two types of network constraints impose a common structure on the probability distribution $p(u, \hat{x}|y; \gamma)$ induced by any admissible multi-stage strategy $\gamma \in \Gamma$, which ultimately determines the associated penalty $J(\gamma)$ given the distribution $p(x, y)$ and costs $c(\hat{x}, x) + \lambda c(u, x)$. To treat these structures in a unified framework, the following subsections will introduce a number of notational conventions along the way. To illustrate their nature here, consider the finite set $\mathcal{Z} = \mathcal{Z}_1 \times \cdots \times \mathcal{Z}_n$, each component set $\mathcal{Z}_i$ representing all symbols received by node i. We let z_i^t, taking values in a finite set $\mathcal{Z}_i^t$, represent the symbols received by node i between making decision u_i^{t-1} and its stage-t decision. Note that there are always a total of $T + 1$ stages, but that only the first T such stages involve communication-related decisions. Nevertheless, it is convenient to define $\mathcal{Z}_i = \mathcal{Z}_i^1 \times \mathcal{Z}_i^2 \times \cdots \times \mathcal{Z}_i^{T+1}$ because the specific T stages in

which the received symbols are nonempty will depend upon the type of network.

■ 5.2.1 Undirected Network Topologies

Assume network topology $\mathcal{F}$ is an n-node undirected graph. The multi-stage communication architecture is taken to be repeated parallel (or synchronous) symbol exchanges between each node i and its immediate neighbors $ne(i)$ in $\mathcal{F}$. In the first stage, every node i generates its decision u_i^1 as a function of only the local measurement y_i. In each subsequent communication stage $t = 2, 3, \ldots, T$, we let z_i^t denote the symbols received by node i, taking values in a finite set $\mathcal{Z}_i^t$. Similarly, we let $z_i^{T+1} \in \mathcal{Z}_i^{T+1}$ denote the symbols received by node i in the final state-related decision stage. It follows that the collection of received symbols z takes its values in a finite set $\mathcal{Z} = \mathcal{Z}_1 \times \cdots \times \mathcal{Z}_n$, each such $\mathcal{Z}_i = \mathcal{Z}_i^2 \times \cdots \times \mathcal{Z}_i^{T+1}$. As is the case for each set $\mathcal{U}_i^t$, the actual cardinality of each set $\mathcal{Z}_i^t$ is exactly as was described for the single-stage architecture in previous chapters, reflecting the local channel model (e.g., erasure versus interference) of node i. Namely, for each $t = 2, 3, \ldots, T+1$, the received information Z_i^t as a function of its neighbors preceding transmissions $u_{ne(i)}^{t-1} = \{u_j^{t-1}; j \in ne(i)\}$ is defined by a conditional distribution $p(z_i^t | x, y, u_{ne(i)}^{t-1})$.

A key opportunity associated with multi-stage processing is the use of online *memory*, which local to each node can include (at most) the received and transmitted symbols in all preceding stages. We denote by $\mathcal{M}_i^t$ the set of all stage-t communication rules local to node i, each of the form

$$\mu_i^t : \mathcal{Y}_i \to \mathcal{U}_i^t$$

when $t = 1$ and, otherwise, each of the form

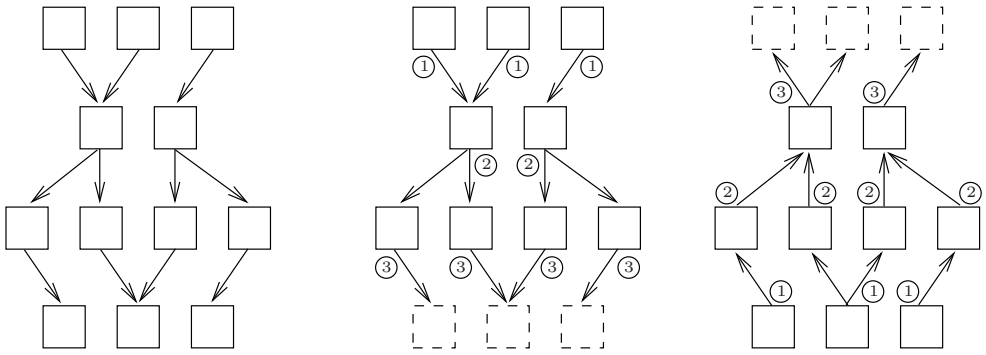
$$\mu_i^t : \mathcal{Y}_i \times \mathcal{U}_i^1 \times \mathcal{Z}_i^2 \times \cdots \times \mathcal{U}_i^{t-1} \times \mathcal{Z}_i^t \to \mathcal{U}_i^t.$$

Similarly, we denote by Δ_i the set of all state-related decision rules local to node i, each of the form

$$\delta_i : \mathcal{Y}_i \times \mathcal{U}_i^1 \times \mathcal{Z}_i^2 \times \cdots \times \mathcal{U}_i^T \times \mathcal{Z}_i^{T+1} \to \mathcal{X}_i.$$

■ 5.2.2 Directed Network Topologies

Assume network topology $\mathcal{F}$ is a directed acyclic graph, where we denote the parents and children of each node i by $pa(i)$ and $ch(i)$, respectively. The multi-stage communication architecture is taken to be repeated forward-backward sweeps on $\mathcal{F}$, a forward sweep for each odd-numbered stage and a backward sweep for each even-numbered stage. Specifically, we follow the convention, illustrated in Figure 5.1, in which each



(a) Directed Network Topology (b) Odd-Numbered Stage (c) Even-Numbered Stage

Figure 5.1. Illustration of the multi-stage communication architecture given (a) a particular directed acyclic network topology, alternating between (b) forward sweeps and (c) reverse sweeps with respect to the partial order (labeled by the circled numbers) implied by the given network. In each odd-numbered (even-numbered) stage t, our convention is to associate the processing rules of all childless (parentless) nodes with the subsequent even-numbered (odd-numbered) stage, as indicated by the dashed nodes.

odd-numbered (even-numbered) communication stage begins with the local processing rules at parentless (childless) nodes in $\mathcal{F}$. As such, if a node is parentless, then a communication rule exists only for odd-numbered stages $t = 1, 3, \ldots, 2\left\lfloor \frac{T-1}{2} \right\rfloor + 1$; if a node is childless, then a communication rule exists only for even-numbered stages $t = 2, 4, \ldots, 2\left\lceil \frac{T-1}{2} \right\rceil$; for every other node, a communication rule exists for every stage $t = 1, 2, \ldots, T$.

Essentially the same notation used in the preceding subsection applies to the case of a directed network topology $\mathcal{F}$. There will, however, be different sets $\mathcal{U}_i^t$ and $\mathcal{Z}_i^t$ in accordance with the per-stage unidirectional communications. In particular, the pivoting roles of parentless/childless nodes between consecutive stages implies that, for each parentless (childless) node i, the sets $\mathcal{U}_i^t$ and $\mathcal{Z}_i^t$ are empty for even-numbered (odd-numbered) stages $t \leq T$. Also in contrast to the multi-stage undirected architecture, the set $\mathcal{Z}_i^1$ is nonempty for every node i that is neither parentless nor childless, while $\mathcal{Z}_i^{T+1}$ is empty for every node i. Moreover, for every (i, t) pair such that the set $\mathcal{Z}_i^t$ is nonempty, we describe the associated channel model by conditional distribution $p(z_i^t | x, y, u_{pa(i,t)})$, defining the input symbols $u_{pa(i,t)}$ by

$$u_{pa(i,t)} = \begin{cases} u_{pa(i)}^t & , \quad t \text{ odd and node } i \text{ is neither parentless nor childless} \\ u_{pa(i)}^{t-1} & , \quad t \text{ even and node } i \text{ is childless} \\ u_{ch(i)}^t & , \quad t \text{ even and node } i \text{ is neither parentless nor childless} \\ u_{ch(i)}^{t-1} & , \quad t \text{ odd and node } i \text{ is parentless} \end{cases} .$$

Similarly, the conventions of Figure 5.1 imply that, for each parentless (childless) node i, the sets $\mathcal{M}_i^t$ are empty for all even-numbered (odd-numbered) stages $t \leq T$. These particular nodes act as pivots within the alternating forward-backward sweeps, parentless nodes initiating every odd-numbered forward sweep and childless nodes initiating every even-numbered backward sweep. Every other node, having both at least one parent and at least one child, makes a decision in every stage. More precisely, for each parentless node i and odd-numbered stage $t \leq T$, the set $\mathcal{M}_i^t$ consists of all rules

$$\mu_i^t : \mathcal{Y}_i \times \mathcal{U}_i^1 \times \mathcal{Z}_i^3 \times \mathcal{U}_i^3 \times \mathcal{Z}_i^5 \cdots \times \mathcal{U}_i^{t-2} \times \mathcal{Z}_i^t \to \mathcal{U}_i^t,$$

while the set Δ_i consists of all rules

$$\delta_i : \mathcal{Y}_i \times \mathcal{U}_i^1 \times \mathcal{Z}_i^3 \times \mathcal{U}_i^3 \times \mathcal{Z}_i^5 \times \cdots \times \mathcal{U}_i^{T-2} \times \mathcal{Z}_i^T \times \mathcal{U}_i^T \to \mathcal{X}_i$$

when T is odd or

$$\delta_i : \mathcal{Y}_i \times \mathcal{U}_i^1 \times \mathcal{Z}_i^3 \times \mathcal{U}_i^3 \times \mathcal{Z}_i^5 \times \cdots \times \mathcal{U}_i^{T-3} \times \mathcal{Z}_i^{T-1} \times \mathcal{U}_i^{T-1} \times \mathcal{Z}_i^T \to \mathcal{X}_i$$

when T is even. For each childless node i and even-numbered stage $t \leq T$, the set $\mathcal{M}_i^t$ consist of all rules

$$\mu_i^t : \mathcal{Y}_i \times Z_i^2 \times U_i^2 \times Z_i^4 \times \cdots \times \mathcal{U}_i^{t-2} \times \mathcal{Z}_i^t \to \mathcal{U}_i^t,$$

while the set Δ_i consists of all rules

$$\delta_i : \mathcal{Y}_i \times \mathcal{Z}_i^2 \times \mathcal{U}_i^2 \times \mathcal{Z}_i^4 \times \mathcal{U}_i^4 \times \cdots \times \mathcal{Z}_i^{T-1} \times \mathcal{U}_i^{T-1} \times \mathcal{Z}_i^T \to \mathcal{X}_i$$

when T is odd or

$$\delta_i : \mathcal{Y}_i \times \mathcal{Z}_i^2 \times \mathcal{U}_i^2 \times \mathcal{Z}_i^4 \times \mathcal{U}_i^4 \times \cdots \times \mathcal{Z}_i^T \times \mathcal{U}_i^T \to \mathcal{X}_i$$

when T is even. Finally, for every other node i, the set $\mathcal{M}_i^t$ for every stage t consists of all rules

$$\mu_i^t : \mathcal{Y}_i \times \mathcal{Z}_i^1 \times \mathcal{U}_i^1 \times \mathcal{Z}_i^2 \cdots \times \mathcal{U}_i^{t-1} \times \mathcal{Z}_i^t \to \mathcal{U}_i^t,$$

while the set Δ_i consists of all rules

$$\delta_i : \mathcal{Y}_i \times \mathcal{Z}_i^1 \times \mathcal{U}_i^1 \times \mathcal{Z}_i^2 \cdots \times \mathcal{U}_i^{T-1} \times \mathcal{Z}_i^T \times \mathcal{U}_i^T \to \mathcal{X}_i.$$

■ 5.2.3 Multi-Stage Probabilistic Structure

We first introduce additional notation and conventions to express the network dependence of a multi-stage architecture, as detailed in the preceding two subsections, in a common mathematical framework. For every node-stage pair (i,t) such that set $\mathcal{Z}_i^t$ is nonempty, denote the input symbols to the associated channel model by

$$u_{tr(i,t)} = \begin{cases} u_{ne(i)}^{t-1} & , \text{ network topology } \mathcal{F} \text{ is undirected} \\ u_{pa(i,t)} & , \text{ network topology } \mathcal{F} \text{ is directed} \end{cases} .$$

Furthermore, for every node-stage pair such that set $\mathcal{Z}_i^t$ is in fact empty, we will model the associated lack of received information as receiving an informationless constant e.g.,

$$p(z_i^t|x,y,u_{tr(i,t)}) = \begin{cases} 1 & , \quad z_i^t = 0 \\ 0 & , \quad \text{otherwise} \end{cases} ,$$

in which case the product

$$\prod_{t=1}^{T+1} p(z_i^t|x,y,u_{tr(i,t)})$$

is, no matter the type of network, equivalent to the product of all single-stage channel models local to node i.

Assumption 5.1 (Memoryless Local Channels)**.** *Local to each node i, every single-stage channel use is independent, conditioned on X and Y, of all channel uses in the preceding stages i.e., for each i, we have*

$$p(z_i|x,y,u_{tr(i)}) = \prod_{t=1}^{T+1} p(z_i^t|x,y,u_{tr(i,t)}),$$

where $u_{tr(i)} = \{u_{tr(i,t)}; t = 1,2,\ldots,T+1\}$ denotes the collection of symbols transmitted to node i over all communication stages.

It is convenient to view the expanding online memory at each node i as the sequential realization of a (local) *information vector*, defined over successive stages $t = 1,\ldots,T+1$ by the recursion

$$I_i^t = \begin{cases} \emptyset & , \quad t = 1 \\ (I_i^{t-1}, z_i^{t-1}, u_i^{t-1}) & , \quad t = 2,\ldots T+1 \end{cases} .$$

Here, we take for granted the network-dependent bookkeeping associated with whether sets $\mathcal{Z}_i^t$ and $\mathcal{U}_i^t$ are in fact empty. We may thus, for each node i, concisely write

$U_i^t = \mu_i^t(Y_i, I_i^t, Z_i^t)$ to denote its t^{th} communication decision and $\hat{X}_i = \delta_i(Y_i, I_i^{T+1}, Z_i^{T+1})$ to denote its final detection decision. By construction, fixing a stage-t communication rule $\mu_i^t \in \mathcal{M}_i^t$ or final detection rule $\delta_i \in \Delta_i$ is equivalent to specifying a distribution

$$p(u_i^t|y_i, I_i^t, z_i^t; \mu_i^t) = \begin{cases} 1 & , \quad \text{if } u_i^t = \mu_i^t(y_i, I_i^t, z_i^t) \\ 0 & , \quad \text{otherwise} \end{cases}$$

or

$$p(\hat{x}_i|y_i, I_i^{T+1}, z_i^{T+1}; \delta_i) = \begin{cases} 1 & , \quad \text{if } \hat{x}_i = \delta_i(y_i, I_i^{T+1}, z_i^{T+1}) \\ 0 & , \quad \text{otherwise} \end{cases} ,$$

respectively.

In either type of network topology, define the set of all admissible *multi-stage* rules local to node i, each a particular sequence of single-stage rules $\gamma_i = (\mu_i^1, \ldots, \mu_i^T, \delta_i)$, by the set $\Gamma_i = \mathcal{M}_i^1 \times \cdots \times \mathcal{M}_i^T \times \Delta_i$. In turn, the set of all admissible multi-stage strategies, each a particular collection $\gamma = (\gamma_1, \ldots, \gamma_n)$ of multi-stage rules, is defined by $\Gamma = \Gamma_1 \times \cdots \times \Gamma_n$. As was the case in the single-stage architectures studied in preceding chapters, the network constraints inherent to any admissible strategy $\gamma \in \Gamma$ induces special probabilistic structure. More precisely, fixing a multi-stage rule $\gamma_i \in \Gamma_i$ is equivalent to specifying the distribution

$$p(u_i, \hat{x}_i|y_i, z_i; \gamma_i) = p(\hat{x}_i|y_i, I_i^{T+1}, z_i^{T+1}; \delta_i) \prod_{t=1}^{T} p(u_i^t|y_i, I_i^t, z_i^t; \mu_i^t), \tag{5.1}$$

which upon incorporating the multi-stage channel model local to node i yields

$$p(u_i, \hat{x}_i|x, y, u_{tr(i)}; \gamma_i) = \sum_{z_i \in \mathcal{Z}_i} p(z_i|x, y, u_{tr(i)}) p(u_i, \hat{x}_i|y_i, z_i; \gamma_i). \tag{5.2}$$

It follows that, for any candidate strategy $\gamma \in \Gamma$,

$$p(u, \hat{x}|x; \gamma) = \int_{y \in \mathcal{Y}} p(y|x) \prod_{i=1}^{n} p(u_i, \hat{x}_i|x, y, u_{tr(i)}; \gamma_i)\, dy \tag{5.3}$$

and the strategy-dependent distribution $p(u, \hat{x}, x; \gamma)$ underlying (3.2) is simply its product with $p(x)$.

■ 5.3 Team-Theoretic Analysis

This section analyzes the multi-stage problem formulation presented in Section 5.2, starting from essentially the same team-theoretic approximations made in the preceding chapters. Recognizing the apparent structural similarities between the strategy-dependent distribution in (5.3) and that of the single-stage undirected architecture in

(4.1), we impose similar simplifying assumptions as in previous chapters i.e., conditional independence, separable cost and measurement/channel/cost locality. We are able to obtain an analytical simplification for the optimal detection strategy $\delta^* \in \Delta = \Delta_1 \times \cdots \times \Delta_n$, showing each δ_i^* lies in a finitely-parameterized subspace of the function space Δ_i. While this parameterization scales linearly in the number of nodes n, it turns out to scale exponentially with the number of stages T. However, we have been unable to analytically deduce the team-optimal communication strategy $\mu^* \in \mathcal{M} = \mathcal{M}_1 \times \cdots \times \mathcal{M}_n$, keeping open the theoretical question of whether it even admits a finite parameterization. Turning to a more pragmatic approach, we offer a conjecture about the form of μ^* based on the known form of δ^*, providing the basis of an approximate solution we describe in Section 5.4.

■ 5.3.1 Necessary Optimality Conditions

As in previous chapters, we start with the conditional independence assumption, which preserves the factorization over nodes i in (5.3) even after marginalizing over the processes Y and Z.

Assumption 5.2 (Conditional Independence)**.** *For every node i,*

$$p(y_i, z_i|x, y_{\setminus i}, z_{\setminus i}, u_{\setminus i}) = p(y_i|x)p(z_i|x, u_{tr(i)}).$$

Lemma 5.1 (Factored Global Representation)**.** *Let Assumption 5.2 hold. For every multi-stage strategy $\gamma \in \Gamma$, the distribution in (5.3) specializes to*

$$p(u, \hat{x}|x; \gamma) = \prod_{i=1}^{n} p(u_i, \hat{x}_i|x, u_{tr(i)}; \gamma_i), \tag{5.4}$$

where for every i

$$p(u_i, \hat{x}_i|x, u_{tr(i)}; \gamma_i) = \sum_{z_i \in \mathcal{Z}_i} p(z_i|x, u_{tr(i)}) \int_{y_i \in \mathcal{Y}_i} p(y_i|x)p(u_i, \hat{x}_i|y_i, z_i; \gamma_i)\, dy_i. \tag{5.5}$$

Proof. With Assumption 5.2 in effect, we may substitute the identity

$$p(z_i|x, y, u_{tr(i)}) = p(z_i|x, u_{tr(i)})$$

into (5.2) and conclude that

$$p(u_i, \hat{x}_i|x, y, u_{tr(i)}; \gamma_i) = \sum_{z_i \in \mathcal{Z}_i} p(z_i|x, u_{tr(i)})p(u_i, \hat{x}_i|y_i, z_i; \gamma_i).$$

It follows that we may substitute the identity

$$p(u_i, \hat{x}_i|x, y, u_{tr(i)}; \gamma_i) = p(u_i, \hat{x}_i|x, y_i, u_{tr(i)}; \gamma_i)$$

for each i into (5.3). Because Assumption 5.2 also implies that $p(y|x) = \prod_i p(y_i|x)$, the integration over $\mathcal{Y}$ can be carried out component-wise i.e.,

$$p(u, \hat{x}, x; \gamma) = p(x) \prod_{i=1}^{n} \int_{y_i \in \mathcal{Y}_i} p(y_i|x) p(u_i, \hat{x}_i|x, y_i, u_{tr(i)}; \gamma_i)\, dy_i.$$

□

The factorization with respect to nodes $i = 1, \ldots, n$ in Lemma 5.1 is a direct consequence of Assumption 5.2 along with the constraints that every node may communicate only with its immediate neighbors in the network. It may at first seem counter-intuitive, in light of the causal multi-stage online processing model, that (5.4) does not also exhibit an explicit factorization with respect to stages $t = 1, \ldots, T$. The caveat is that these successive decision stages collectively operate on the same measurement vector $Y = y$. It is rather the side information local to each node i that grows over successive stages, providing the increasingly global context in which to *reprocess* the local measurement $Y_i = y_i$. Nonetheless, the causal processing can be exploited to simplify the (offline) local marginalization in (5.5) associated with fixing a local multi-stage rule $\gamma_i \in \Gamma_i$. In particular, given Assumption 5.2 holds, each node i may firstly decompose the integral over all of $\mathcal{Y}_i$ into a finite collection of integrals over memory-dependent sub-regions of $\mathcal{Y}_i$ and secondly, given Assumption 5.1 also holds, evaluate the sum over $\mathcal{Z}_i$ in a recursive fashion.

The following lemmas formalize this structure in the (offline) local computation at each node i, which will require yet more notation. What needs to expressed is a sequential paring down of the measurement space by the successive communication stages. Designing the stage-one rule μ_i^1 local to node i is essentially the same as that of a single-stage architecture, i.e., we seek a specific partition of the local measurement space $\mathcal{Y}_i$ into $|\mathcal{U}_i^1|$ decision regions, one such partition per value of z_i^1. It follows that, upon fixing the stage-one rule μ_i^1, the realization of a specific value of stage-two memory $I_i^2 = (z_i^1, u_i^1)$ implies that local measurement y_i must lie in the specific subset of $\mathcal{Y}_i$ for which $u_i^1 = \mu_i^1(y_i, z_i^1)$. Designing the stage-two rule is similarly equated with selecting a collection of size-$|\mathcal{U}_i^2|$ partitions, but restricted to a different subset of $\mathcal{Y}_i$ as a function of the assumed value of memory I_i^2 and the fixed stage-one rule μ_i^1. More generally, let $u_i^{s:s'}$ denote a subsequence of communication decisions taking values in the set $\mathcal{U}_i^s \times$

$\mathcal{U}_i^{s+1} \times \cdots \times \mathcal{U}_i^{s'}$. Similarly, $u_{tr(i)}^{s:s'}$, $z_i^{s:s'}$ or $\mu_i^{s:s'}$ denote such a sequence of channel symbols or local communication rules, respectively. Consider a node i and assume multi-stage rule $\gamma_i = (\mu_i^{1:T}, \delta_i)$ is fixed: for every stage $t = 1, \ldots, T$, the set $\mathcal{Y}_i(I_i^{t+1}; \mu_i^{1:t})$ denotes the subset of $\mathcal{Y}_i$ for which $u_i^s = \mu_i^s(y_i, I_i^s, z_i^s)$ in every stage $s \leq t$. The subset $\mathcal{Y}_i(I_i^{T+2}; \gamma_i)$ is analogously defined, including all $T+1$ decision stages and the identity $I_i^{T+2} = (I_i^{T+1}, z_i^{T+1}, \hat{x}_i)$. Recognizing that the memory $I_i^{t+1} \supset I_i^t$ expands with each additional stage, it follows that the subsets $\mathcal{Y}_i(I_i^{t+1}; \mu_i^{1:t}) \subset \mathcal{Y}_i(I_i^t; \mu_i^{1:t-1})$ shrink with each additional stage.

Lemma 5.2 (Factored Local Representation)**.** *Let Assumption 5.2 hold. For any fixed multi-stage rule $\gamma_i \in \Gamma_i$ local to node i, we have*

$$p(u_i, \hat{x}_i | x, z_i; \gamma_i) = p(\hat{x}_i | x, I_i^{T+1}, z_i^{T+1}; \gamma_i) \prod_{t=1}^{T} p(u_i^t | x, I_i^t, z_i^t; \mu_i^{1:t}) \tag{5.6}$$

with

$$p(\hat{x}_i | x, I_i^{T+1}, z_i^{T+1}; \gamma_i) = \frac{\mathbf{P}\left[Y_i \in \mathcal{Y}_i(I_i^{T+2}; \gamma_i) \middle| X = x \right]}{\mathbf{P}\left[Y_i \in \mathcal{Y}_i(I_i^{T+1}; \mu_i) \middle| X = x \right]}$$

and

$$p(u_i^t | x, I_i^t, z_i^t; \mu_i^{1:t}) = \frac{\mathbf{P}\left[Y_i \in \mathcal{Y}_i(I_i^{t+1}; \mu_i^{1:t}) \middle| X = x \right]}{\mathbf{P}\left[Y_i \in \mathcal{Y}_i(I_i^t; \mu_i^{1:t-1}) \middle| X = x \right]}, \qquad t = 1, 2, \ldots, T.$$

Proof. Firstly, Assumption 5.2 implies that, no matter the fixed strategy γ, process Y_i and Z_i are conditionally independent given X, which follows from

$$\begin{aligned} p(y_i, z_i | x; \gamma) &= \int_{y_{\setminus i}} \sum_{z_{\setminus i}} \sum_{u_{\setminus i}} p(y_{\setminus i}, z_{\setminus i}, u_{\setminus i} | x; \gamma) p(y_i, z_i | x, y_{\setminus i}, z_{\setminus i}, u_{\setminus i}) \, dy_{\setminus i} \\ &= \sum_{u_{tr(i)}} p(u_{tr(i)} | x; \gamma) p(y_i | x) p(z_i | x, u_{tr(i)}) \;=\; p(y_i | x) p(z_i | x; \gamma). \end{aligned}$$

Next, express $p(u_i, \hat{x}_i | x, z_i; \gamma_i)$ as the product $p(\hat{x}_i | x, z_i, u_i; \gamma_i) p(u_i | x, z_i; \gamma_i)$, then similarly express $p(u_i | x, z_i; \gamma_i)$ as the product $p(u_i^T | x, z_i, u_i^{1:T-1}; \gamma_i) p(u_i^{1:T-1} | x, z_i; \gamma_i)$ and so on until we obtain the identity

$$p(u_i, \hat{x}_i | x, z_i; \gamma_i) = p(\hat{x}_i | x, z_i, u_i; \gamma_i) \prod_{t=1}^{T} p(u_i^t | x, z_i, u_i^{1:t-1}; \gamma_i).$$

Consider the factor for stage $t = 1$ (where I_i^1 is empty by definition), which must itself satisfy the identity

$$p(u_i^1|x, z_i; \gamma_i) = \int_{y_i \in \mathcal{Y}_i} p(y_i|x, z_i; \gamma_i) p(u_i^1|x, y_i, z_i; \gamma_i)\, dy_i.$$

We've already concluded that $p(y_i|x, z_i; \gamma_i) = p(y_i|x)$ and, for any fixed stage-one communication rule μ_i^1, we have $p(u_i^1|x, y_i, z_i; \gamma_i) = p(u_i^1|y_i, I_i^1, z_i^1; \mu_i^1)$, so that

$$\begin{aligned} p(u_i^1|x, z_i; \gamma_i) &= \int_{y_i \in \mathcal{Y}_i} p(y_i|x) p(u_i^1|y_i, z_i^1; \mu_i^1)\, dy_i \\ &= \mathbf{P}\left[\mu_i^1(Y_i, I_i^1, z_i^1) = u_i^1 \middle| X = x\right] = \mathbf{P}\left[Y_i \in \mathcal{Y}_i(I_i^2; \mu_i^1) \middle| X = x\right] \\ &\equiv p(u_i^1|x, I_i^1, z_i^1; \mu_i^1). \end{aligned}$$

For stage $t = 2$, the same basic steps lead to the identity

$$p(u_i^2|x, z_i, u_i^1; \gamma_i) = \int_{y_i \in \mathcal{Y}_i} p(y_i|x, z_i, u_i^1; \gamma_i) p(u_i^2|y_i, I_i^2, z_i^2; \mu_i^2)\, dy_i.$$

Again appealing to the definition of the multi-stage rule γ_i, we may write

$$\begin{aligned} p(y_i|x, z_i, u_i^1; \gamma_i) &= \frac{p(y_i|x, z_i; \gamma_i) p(u_i^1|x, y_i, z_i; \gamma_i)}{p(u_i^1|x, z_i; \gamma_i)} \\ &= \frac{p(y_i|x) p(u_i^1|y_i, I_i^1, z_i^1; \mu_i^1)}{p(u_i^1|x, I_i^1, z_i^1; \mu_i^1)} \\ &= \begin{cases} \frac{p(y_i|x)}{\mathbf{P}\left[Y_i \in \mathcal{Y}_i(I_i^2; \mu_i^1) \middle| X = x\right]} & , \text{ if } y_i \in \mathcal{Y}_i(I_i^2; \mu_i^1) \\ 0 & , \text{ otherwise} \end{cases} \\ &\equiv p(y_i|x, I_i^2; \mu_i^1) \end{aligned}$$

and, also recognizing that $I_i^3 \supset I_i^2$ and so $\mathcal{Y}_i(I_i^3; \mu_i^{1:2}) \subset \mathcal{Y}_i(I_i^2; \mu_i^1)$, we have

$$\begin{aligned} p(u_i^2|x, z_i, u_i^1; \gamma_i) &= \int_{y_i \in \mathcal{Y}_i(I_i^2; \mu_i^1)} \frac{p(y_i|x) p(u_i^2|y_i, I_i^2, z_i^2; \mu_i^2)}{\mathbf{P}\left[Y_i \in \mathcal{Y}_i(I_i^2; \mu_i^1) \middle| X = x\right]}\, dy_i \\ &= \frac{\mathbf{P}\left[Y_i \in \mathcal{Y}_i(I_i^3; \mu_i^{1:2}) \middle| X = x\right]}{\mathbf{P}\left[Y_i \in \mathcal{Y}_i(I_i^2; \mu_i^1) \middle| X = x\right]} \\ &\equiv p(u_i^2|x, I_i^2, z_i^2; \mu_i^{1:2}). \end{aligned}$$

Continuing the induction, we conclude for every stage $t \leq T$ that

$$p(u_i^t|x, z_i, u_i^{1:t-1}; \gamma_i) = \int_{y_i \in \mathcal{Y}_i} p(y_i|x, z_i, u_i^{1:t-1}; \gamma_i) p(u_i^t|y_i, I_i^t, z_i^t; \mu_i^t)\, dy_i$$

with

$$\begin{aligned} p(y_i|x, z_i, u_i^{1:t-1}; \gamma_i) &= \frac{p(y_i|x) p(u_i^{1:t-1}|y_i, z_i; \gamma_i)}{p(u_i^{1:t-1}|x, z_i; \gamma_i)} \\ &= \begin{cases} \frac{p(y_i|x)}{\mathbf{P}\left[Y_i \in \mathcal{Y}_i(I_i^t; \mu_i^{1:t-1}) \middle| X = x\right]} & , \text{ if } y_i \in \mathcal{Y}_i(I_i^t; \mu_i^{1:t-1}) \\ 0 & , \text{ otherwise} \end{cases} \\ &\equiv p(y_i|x, I_i^t; \mu_i^{1:t-1}) \end{aligned}$$

and, in turn,

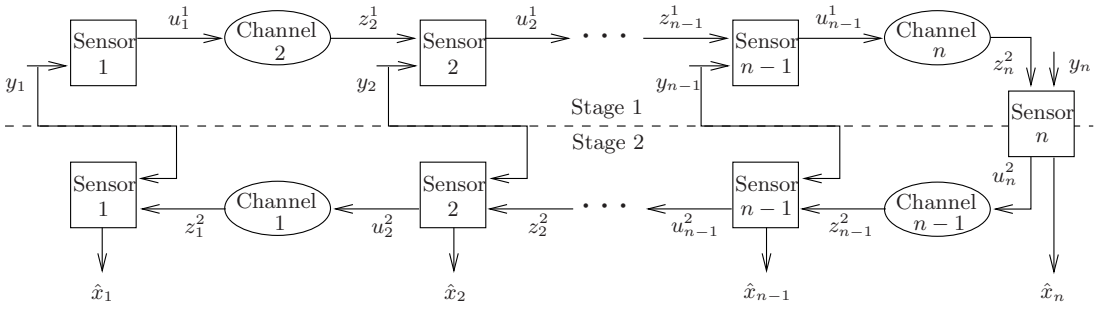
$$\begin{aligned} p(u_i^t|x, z_i, u_i^{1:t-1}; \gamma_i) &= \frac{\mathbf{P}\left[Y_i \in \mathcal{Y}_i(I_i^{t+1}; \mu_i^{1:t}) \middle| X = x\right]}{\mathbf{P}\left[Y_i \in \mathcal{Y}_i(I_i^t; \mu_i^{1:t-1}) \middle| X = x\right]} \\ &\equiv p(u_i^t|x, I_i^t, z_i^t; \mu_i^{1:t}). \end{aligned}$$

The exact same arguments apply to the final decision stage $t = T + 1$, involving $\hat{x}_i = \delta_i(y_i, I_i^{T+1}, z_i^{T+1})$. □

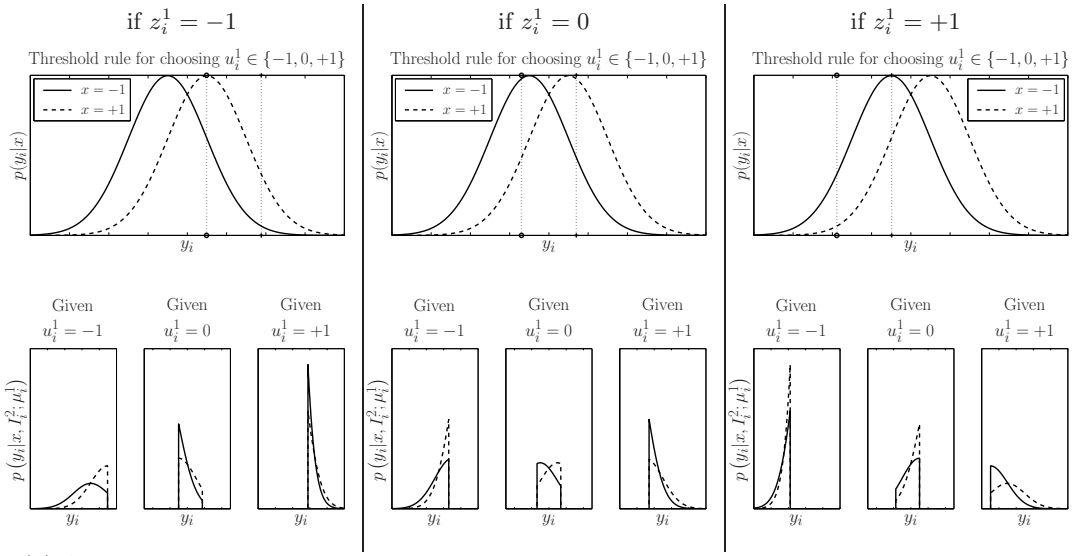
Lemma 5.2 reveals much about the (online and offline) processing requirements local to each node i in a multi-stage communication architecture. From the online perspective, each node uses its memory to sequentially pare down the local likelihood i.e., the proof to Lemma 5.2 established that

$$p(y_i|x, I_i^t; \gamma_i) \propto \begin{cases} p(y_i|x) & , \quad y_i \in \mathcal{Y}_i[I_i^t; \mu_i^{1:t-1}] \\ 0 & , \text{ otherwise} \end{cases} \tag{5.7}$$

with $\mathcal{Y}_i(I_i^t; \mu_i^{1:t-1}) \supset \mathcal{Y}_i(I_i^{t-1}; \mu_i^{1:t-2})$ for every stage t. Figure 5.2 illustrates the first stage of this memory-dependent likelihood evolution for the special case of a directed tandem network with a global binary state and n linear Gaussian detectors. The same trend continues with each additional stage: every node essentially hones in on a smallest subregion over which it must make use of the likelihood vector $p(y_i|x)$, doing so in each stage t as a function of the expanding information vector I_i^t and the preceding communication rules $\mu_i^{1:t-1}$.



(a) Two-Stage ($T = 2$) Online Processing Model in an n-Node Directed Series Network



(b) A Fixed Stage-One Communication Rule and the Memory-Dependent Stage-Two Likelihood

Figure 5.2. Illustration of the memory-dependent paring down of local likelihoods suggested by the probabilistic structure exposed in Lemma 5.2. In the directed network shown in (a), we assume each node i employs the first-stage communication rule $u_i^1 = \mu_i^1(yi, z_i^1)$ using the thresholds shown in the top row of (b). (Strictly-speaking, only the middle column of (b) applies for node 1). The second row of (b) shows the second-stage likelihood functions for different realizations of local memory $I_i^2 = (z_i^1, u_i^1)$, each proportional to the original likelihood $p(y_i|x)$ over a particular subinterval in $\mathbb{R}$ and otherwise zero.

From the offline perspective, note that Lemma 5.2 specializes (5.5) to

$$p(u_i, \hat{x}_i|x, u_{tr(i)}; \gamma_i) = \sum_{z_i \in \mathcal{Z}_i} p(z_i|x, u_{tr(i)}) p(u_i, \hat{x}_i|x, z_i; \gamma_i)$$

with

$$p(u_i, \hat{x}_i|x, z_i; \gamma_i) = \mathbf{P}\left[Y_i \in \mathcal{Y}_i(I_i^{T+2}; \gamma_i) \middle| X = x \right] = \int_{y_i \in \mathcal{Y}_i(I_i^{T+1}; \gamma_i)} p(y_i|x)\, dy_i.$$

The latter equation reveals that each node i can marginalize over Y_i in a piece-meal fashion: first, partition the measurement space $\mathcal{Y}_i$ into $|\mathcal{U}_i \times \mathcal{X}_i \times \mathcal{Z}_i|$ disjoint subsets, one component per possible realization of the local information vector I_i^{T+2}; then, compute the event probabilities associated with Y_i (conditioned on $X = x$ for each $x \in \mathcal{X}$) lying in the subsets $\mathcal{Y}_i(I_i^{T+2}; \gamma_i)$ corresponding to this partition. In other words, fixing the rule of each successive decision stage confines the probabilistic support of Y_i to a successively smaller subset of its original space $\mathcal{Y}_i$ as a function of the expanding information vector local to node i.

The next lemma describes an additional simplification in each node's (offline) local computation when its multi-stage channel model is also memoryless.

Lemma 5.3 (Recursive Local Marginalization)**.** *Let Assumption 5.1 and Assumption 5.2 both hold. For any fixed multi-stage rule $\gamma_i \in \Gamma_i$ local to node i, the associated ith factor $p(u_i, \hat{x}_i | x, u_{tr(i)}; \gamma_i)$ in (5.5) is given by the following recursive definition: initialize*

$$p(\hat{x}_i | x, I_i^{T+1}, u_{tr(i,T+1)}; \gamma_i) = \sum_{z_i^{T+1} \in \mathcal{Z}_i^{T+1}} p(z_i^{T+1} | x, u_{tr(i,T+1)}) p(\hat{x}_i | x, I_i^{T+1}, z_i^{T+1}; \gamma_i)$$

and then, for $t = T, T-1, \ldots, 1$, evaluate

$$p(u_i^{t:T}, \hat{x}_i | x, I^t, u_{tr(i)}^{t:T+1}; \gamma_i) = \sum_{z_i^t \in \mathcal{Z}_i^t} p(z_i^t | x, u_{tr(i,t)}) p(u_i^t | x, I_i^t, z_i^t; \mu_i^{1:t-1}) p(u_i^{t+1:T}, \hat{x}_i | x, I^{t+1}, u_{tr(i)}^{t+1:T+1}; \gamma_i) \ .$$

Proof. Assumption 5.1 and Assumption 5.2 taken together implies

$$p(z_i | x, y, u_{tr(i)}) = \prod_{t=1}^{T+1} p(z_i^t | x, u_{tr(i,t)}).$$

Substituting this identity and (5.6) into (5.5), we have

$$p(u_i, z_i | x, u_{tr(i)}; \gamma_i) = \sum_{z_i} p(z_i^{T+1} | x, u_{tr(i,T+1)}) p(\hat{x}_i | x, I_i^{T+1}, z_i^{T+1}; \gamma_i) \times \prod_{t=1}^{T} p(z_i^t | x, u_{tr(i,t)}) p(u_i^t | x, I_i^t, z_i^t; \mu_i^{1:t}).$$

We may then distribute the summation over $\mathcal{Z}_i$ through the factors over stages t, yielding exactly the stated recursions. □

We now present the main results of this chapter: under Assumption 5.2, the optimal detection strategy $\delta^* = (\delta_1^*, \ldots, \delta_n^*)$ lies in a finitely-parameterized subspace of $\Delta_1 \times \cdots \times \Delta_n$ and, upon also introducing the usual locality assumptions, this finite parameterization scales linearly with the number of nodes n. As mentioned earlier, we have not been successful in similarly deducing the form of the optimal communication strategy $\mu^* = (\mu_1^*, \ldots, \mu_n^*)$, each ith multi-stage rule $\mu_i^* \in \mathcal{M}_i^1 \times \cdots \times \mathcal{M}_i^T$. The distinct complication that arises in the case of multiple communication stages is that each node's current transmission can impact the information it may receive in future stages. On the receiving side, each node can therefore exploit the context of all past information in order to best interpret the newest symbol of information. In addition, taking into account that every other node is able to do the same, each node aims to generate the most resourceful sequence of transmissions for its neighbors, potentially adapting each communication stage with each successive new symbol of information. In the final detection stage, of course, the incentives for signaling to influence the information in future stages entirely disappears, and it is only the receiving side that remains; yet, even under best-case model assumptions, the form of the final-stage rule reveals that these signaling mechanisms depend jointly, *not* recursively, on the available information from other communication stages. These phenomena lie at the heart of the exponential complexity in T exhibited by the finite parameterization for δ^*. Accordingly, recognizing that each node's communication rule will feature both sides (i.e., receiving and transmitting) to these signaling incentives, we should expect the parameterization (if even finite) for μ^* to also scale exponentially in T.

Proposition 5.1 (Optimal Parameterization of Detection Stage)**.** *Let Assumption 5.2 hold. Consider any particular node i and assume that both the local communication rules and the multi-stage rules local to all other nodes are fixed at their optimal values, which we denote by $\mu_i^* \in \mathcal{M}_i^1 \times \cdots \times \mathcal{M}_i^T$ and $\gamma_{\setminus i}^* = \{\gamma_j^* \in \Gamma_j \mid j \neq i\}$, respectively. Then, the optimal final-stage detection rule local to node i reduces to*

$$\delta_i^*(Y_i, U_i, Z_i) = \arg\min_{\hat{x}_i \in \mathcal{X}_i} \sum_{x \in \mathcal{X}} b_i^*(\hat{x}_i, x; U_i, Z_i) p(Y_i | x, I_i^{T+1}; \mu_i^*),$$

where parameter values $b_i^* \in \mathbb{R}^{|\mathcal{X}_i \times \mathcal{X} \times \mathcal{U}_i \times \mathcal{Z}_i|}$ depend on all fixed rules according to

$$b_i^*(\hat{x}_i, x; u_i, z_i) = p(x) \sum_{u_{\setminus i}} \sum_{\hat{x}_{\setminus i}} c(u, \hat{x}, x) p(u_i, z_i | x, u_{tr(i)}; \mu_i^*) \prod_{j \neq i} p(u_j, \hat{x}_j | x, u_{tr(j)}; \gamma_j^*)$$

with

$$p(u_i, z_i | x, u_{tr(i)}; \mu_i^*) = p(z_i | x, u_{tr(i)}) \prod_{t=1}^{T} p(u_i^t | x, I_i^t, z_i^t; \mu_i^*).$$

Proof. Notice that the distribution $p(u_i, \hat{x}_i | x, u_{tr(i)}; \gamma_i)$ in Lemma 5.1 is structurally identical to its counterpart for the single-stage undirected architecture, albeit here both u_i and $u_{tr(i)}$ are discrete-valued length-T vectors. We follow essentially the same steps taken in the proof to Proposition 4.1 for the detection rule in the single-stage undirected architecture. See Appendix C.1 for details. □

It is instructive to compare the structure of the optimal detection strategy $\delta^* = (\delta_1^*, \ldots, \delta_n^*)$ in Proposition 5.1 with that in Proposition 4.1 for the single-stage undirected architecture analyzed in the preceding chapter. Here, we see that memory I_i^{T+1} manifests itself in the sufficient statistic for local measurement $Y_i = y_i$, appearing as a conditioning argument to the local likelihood vector $p(y_i | x, I_i^{T+1}; \mu_i^*)$. This expresses the same memory-dependent sequential "paring down" of the local likelihood function revealed in Lemma 5.2 (and illustrated in Figure 5.2).

■ 5.3.2 Efficient Online Computation

As in previous chapters, with respect to the number of nodes n, efficient online computation requires the introduction of certain locality assumptions. We will see, however, that in a multi-stage architecture these assumptions are not sufficient to guarantee efficient offline computation (with respect to n), nor do these assumption alleviate the exponential complexity in the number of stages T.

Assumption 5.3 (Measurement/Channel Locality)**.** *The measurement model and multi-stage channel model local to each node i are independent of all non-local state variables $X_{\setminus i}$ i.e., for every i,*

$$p(y_i|x) = p(y_i|x_i) \qquad \text{and} \qquad p(z_i|x, y, u_{tr(i)}) = p(z_i|x_i, y, u_{tr(i)}).$$

Proposition 5.2 (Detection-Stage Online Efficiency)**.** *If Assumption 5.3 also holds, then Proposition 5.1 specializes to*

$$\delta_i^*(Y_i, U_i, Z_i) = \arg\min_{\hat{x}_i \in \mathcal{X}_i} \sum_{x_i \in \mathcal{X}_i} \beta_i^*(\hat{x}_i, x_i; U_i, Z_i) p(Y_i | x_i, I_i^{T+1}; \mu_i^*)$$

with

$$\beta_i^*(\hat{x}_i, x_i; u_i, z_i) = \sum_{x_{\setminus i}} b_i^*(\hat{x}_i, x; u_i, z_i).$$

Proof. Starting with Proposition 5.1, it suffices to show that the addition of Assumption 5.3 implies

$$p(y_i | x, u_i, z_i; \mu_i^*) = p(y_i | x_i, u_i, z_i; \mu_i^*).$$

Measurement locality in Lemma 5.1 implies that, for any fixed local communication rules μ_i,

$$p(y_i, u_i|x, z_i; \mu_i) = p(y_i|x_i) \prod_{t=1}^{T} p(u_i^t|y_i, I_i^t, z_i^t; \mu_i^t) = p(y_i, u_i|x_i, z_i; \mu_i)$$

and, in turn,

$$\begin{aligned} p(y_i|x, u_i, z_i; \mu_i) &= \frac{p(y_i, u_i|x, z_i; \mu_i)}{\int_{y_i} p(y_i, u_i|x, z_i; \mu_i)\, dy_i} = \frac{p(y_i, u_i|x_i, z_i; \mu_i)}{\int_{y_i} p(y_i, u_i|x_i, z_i; \mu_i)\, dy_i} \\ &= p(y_i|x_i, u_i, z_i; \mu_i). \end{aligned}$$

□

While Proposition 5.2 shows that the optimal detection strategy δ^* admits a finite parameterization $\beta^* = (\beta_1^*, \ldots, \beta_n^*)$ that scales linearly in the number of nodes n, offline computation will surely scale exponentially with n in the absence of special cost structure.

Assumption 5.4 (Cost Locality)**.** *The global communication costs and global detection costs are additive across both nodes and stages, each term local to node i independent of all non-local state variables $X_{\setminus i}$ i.e.,*

$$c(u, x) = \sum_{i=1}^{n} \sum_{t=1}^{T} c(u_i^t, x_i) \qquad \text{and} \qquad c(\hat{x}, x) = \sum_{i=1}^{n} c(\hat{x}_i, x_i). \tag{5.8}$$

Proposition 5.3 (Detection-Stage Offline Computation)**.** *If Assumptions 5.1–5.4 all hold, then Proposition 5.2 applies with rule parameters specialized to the proportionality*

$$\beta_i^*(\hat{x}_i, x_i; u_i, z_i) \propto p(x_i) P_i^*(u_i, z_i|x_i) c(\hat{x}_i, x_i), \tag{5.9}$$

where the (fixed) global communication strategy μ^* determines the likelihood function

$$P_i^*(u_i, z_i|x_i) = \sum_{u_{\setminus i} \in \mathcal{U}_{\setminus i}} p(u_i, z_i|x_i, u_{tr(i)}; \mu_i^*) \sum_{x_{\setminus i} \in \mathcal{X}_{\setminus i}} p(x_{\setminus i}|x_i) \prod_{j \neq i} p(u_j|x_j, u_{tr(j)}^{1:T}; \mu_j^*) \tag{5.10}$$

with the different factors in (5.10) given by

$$p(u_i, z_i|x_i, u_{tr(i)}; \mu_i^*) = p(z_i|x_i, u_{tr(i)}) \prod_{t=1}^{T} p(u_i^t|x_i, I_i^t, z_i^t; \mu_i^*)$$

and, for every $j \neq i$,

$$p(u_j|x_j, u_{tr(j)}^{1:T}; \mu_j^*) = \sum_{z_j \in \mathcal{Z}_j} p(z_j|x_j, u_{tr(j)}) \prod_{t=1}^{T} p(u_j^t|x_j, I_j^t, z_j^t; \mu_j^*).$$

Proof. Starting with Proposition 5.2, we follow essentially the same steps taken in the proof to Proposition 4.2 for the detection rule in the single-stage undirected architecture. See Appendix C.2 for details. □

Proposition 5.3 has a number of important implications about the structure of team-optimal solutions in multi-stage architectures. Firstly, it is instructive to contrast δ^* with the myopic strategy identified in Chapter 2. Each component rule δ_i^* is seen to make two different uses of memory (i.e., its local information vector I_i^{T+1}). The first is in paring down its local measurement likelihoods, exactly as was highlighted in Figure 5.2. The second is in interpreting the symbol vector z_i received over the T preceding stages of online communication, entering as a Bayesian correction (i.e., a reweighting of the prior probabilities by the likelihood P_i^*) to the myopic rule parameters $p(x_i)c(\hat{x}_i, x_i)$ identified in Chapter 2. Equation (5.10) reveals how each such likelihood function P_i^*, depends *jointly* on the local information vector $(u_i, z_i) = (I_i^{T+1}, z_i^{T+1})$. That is, with respect to making the optimal final state-related decision $\hat{x}_i$, each node i must interpret the received information z_i in the full context of the information u_i it transmitted in previous stages. This joint dependence of P_i^* on (z_i, u_i) carries over to the local parameterization β_i^* and, in turn, the full parameter vector β^* scales exponentially with the number of stages T.

Proposition 5.3 also reveals interesting ties to the (sum-product) belief propagation algorithm discussed in Chapter 2. Recall from Example 2.7 that, if for each i we choose $c(\hat{x}_i, x_i)$ equal to unity given $\hat{x}_i \neq x_i$ and zero otherwise, then the optimal *centralized* detector (per joint realization $Y = y$) amounts to each node selecting the mode of its posterior marginal $p(x_i|y) \propto p(x_i, y)$. Thus, assuming the prior probabilities $p(x)$ are defined by a graphical model, belief propagation algorithms become applicable to this (unconstrained) decision problem, essentially yielding at every node an exact (in junction trees) or approximate (in graphs with cycles) sufficient statistic for the global measurement $Y = y$. Let us substitute this specific cost function into Proposition 5.3, yielding for each realization $(Y_i, U_i, Z_i) = (y_i, I_i^{T+1}, z_i^{T+1})$ a final-stage detection rule of the form

$$\hat{x}_i = \delta_i^*(y_i, I_i^{T+1}, z_i^{T+1}) = \arg\max_{x_i \in \mathcal{X}_i} p(x_i) P_i^*(I_i^{T+1}, z_i^{T+1}|x_i) p(y_i|x_i, I_i^{T+1}; \mu_i^*). \quad (5.11)$$

From the proof to Proposition 5.3, we recognize that

$$p(x_i) P_i^*(I_i^{T+1}, z_i^{T+1}|x_i) p(y_i|x_i, I_i^{T+1}; \mu_i^*) \propto p(x_i|y_i, u_i, z_i; \mu^*),$$

and thus δ_i^* can be viewed as selecting the mode of the *network-constrained* posterior marginal $p(x_i|y_i, I_i^{T+1}, z_i^{T+1}; \mu)$. Assuming no explicit communication costs (i.e., assuming parameter $\lambda = 0$), the role of the optimal multi-stage communication strategy $\mu^* \in \mathcal{M}_1 \times \cdots \times \mathcal{M}_n$ similarly specializes: it is to map the global measurement y into the sequence of symbols (u, z) such that every node i may use the accessible portion of those symbols, namely (u_i, z_i), alongside its local measurement y_i to best approximate its (centralized) sufficient statistic $p(x_i|y)$.

Carrying the observed ties to belief propagation one step further, we now use (5.11) to back out a "belief update" equation over successive online communication stages $t = 1, 2, \ldots$. For the moment, let us take offline computation for granted: specifically, we take the T-stage communication strategy $\mu \in \mathcal{M}_1 \times \cdots \times \mathcal{M}_n$ to be fixed, and assume every node i knows the associated likelihood function $P_i^\mu(I_i^{T+1}, z_i^{T+1}|x_i)$ as well as the local prior $p(x_i)$. Recall from (5.10) that

$$P_i^\mu(I_i^{T+1}, z_i^{T+1}|x_i) = p(u_i, z_i|x_i; \mu) = p(z_i^{T+1}|x_i, I_i^{T+1}; \mu)p(I_i^{T+1}|x_i; \mu)$$

and from the same reasoning underlying (5.7) that, for any $y_i \in \mathcal{Y}_i[I_i^{T+1}; \mu_i]$,

$$p(y_i|x_i, I_i^{T+1}; \mu_i) \propto \frac{p(y_i|x_i)}{p(I_i^{T+1}|x_i; \mu)}.$$

In fact, these relationships hold for all $t \leq T+1$, where from Lemma 5.2 we can suitably marginalize the likelihood function P_i^μ to obtain

$$p(z_i^t|x_i, I_i^t; \mu) = \frac{p(I_i^t, z_i^t|x_i; \mu)}{p(I_i^t|x_i; \mu)}$$

while from Lemma 5.3 we obtain

$$p(y_i|x_i, I_i^t; \mu_i) \propto \frac{p(y_i|x_i)}{p(I_i^t|x_i; \mu)}, \qquad y_i \in \mathcal{Y}_i[I_i^t; \mu_i^{1:t-1}].$$

Altogether, our network-constrained analog to the "belief update" equation local to each node i becomes

$$M_i^t(x_i) := p(x_i|y_i, I_i^t, z_i^t; \mu) \propto p(x_i)p(z_i^t|x_i, I_i^t; \mu)p(y_i|x_i), \quad t = 1, 2, \ldots T+1. \quad (5.12)$$

In the two preceding chapters, we found that Assumptions 5.2–5.4 (along with a polytree topology in the case of a directed network $\mathcal{F}$) were sufficient to obtain an efficient message-passing algorithm for computing all nodes' likelihood statistics $\{P_i^*; i \in \mathcal{V}\}$. Proposition 5.3 shows that additional model assumptions will be required in multi-stage architectures, if even an analogously efficient offline message-passing algorithm

for its team-optimal solution exists in the first place. To better appreciate this fact, first note that (by constraint) the subset of the communication decisions $u_{tr(i)}$ that influence the side information Z_i local to node i excludes all non-neighboring nodes' communication decisions i.e., $u_{tr(i)} \subseteq u_{ne(i)} \subset u_{\setminus i}$. It follows that

$$\sum_{u_{\setminus i}} p(u_i, z_i|x_i, u_{tr(i)}; \mu_i^*) \prod_{j \neq i} p(u_j|x_j, u_{tr(j)}^{1:T}; \mu_j^*) = \sum_{u_{ne(i)}} p(u_i, z_i|x_i, u_{tr(i)}; \mu_i^*) \sum_{u_{\mathcal{V}\setminus i - ne(i)}} \prod_{j \neq i} p(u_j|x_j, u_{tr(j)}^{1:T}; \mu_j^*),$$

which upon substitution into (5.10) yields

$$P_i^*(u_i, z_i|x_i) = \sum_{u_{ne(i)}} p(u_i, z_i|x_i, u_{tr(i)}; \mu_i^*) P_{ne(i)\to i}^*(u_{ne(i)}|x_i, u_i),$$

with

$$P_{ne(i)\to i}^*(u_{ne(i)}|x_i, u_i) = \sum_{x_{\setminus i}} p(x_{\setminus i}|x_i) \sum_{u_{\mathcal{V}\setminus i - ne(i)}} \prod_{j \neq i} p(u_j|x_j, u_{tr(j)}^{1:T}; \mu_j^*). \tag{5.13}$$

At each node i, we may view $P_{ne(i)\to i}^*$ as the multi-stage analog to the incoming likelihood messages discussed in previous chapters. In contrast to the single-stage counterparts, however, (5.13) does not readily present itself as a recursive factorization on the network topology $\mathcal{F}$. Indeed, (5.13) suggests that the offline computation to support the optimal detection strategy δ^* scales exponentially with n (as well as T), at least without either (i) specializing the analysis to whether network topology $\mathcal{F}$ is directed or undirected, (ii) introducing more structure on the probability graph $\mathcal{G}$ underlying the global prior $p(x)$ or (iii) some special-case combinations of (i) and (ii). Such pursuits are left for future work.

As commented earlier, we expect the difficulties associated with the team-optimal multi-stage communication strategy μ^* to be even more pronounced than those uncovered for δ^*. In preparation for an approximate offline algorithm we describe next, and then experiment with in Section 5.5, we close this section with a conjecture on the existence of a finite parameterization for μ^*. It is inspired by the efficient offline message-passing algorithms that were derived in the preceding chapters, and its proof (or disproof) is also left for future work.

Conjecture 5.1 (Optimal Parameterization of Communication Stages)**.** *Let Assumptions 5.1–5.4 hold. Assume all rules except for the stage-t communication rule local to node i are fixed at their optimal values. There exist both a likelihood function*

$P_i^t(I_i^t, z_i^t|x_i)$ and a cost function $C_i^t(I_i^{t+1}, x_i)$ such that the optimal rule over all $\mathcal{M}_i^t$ is given by

$$\mu_i^t(Y_i, I_i^t, Z_i^t) = \arg\min_{u_i^t \in \mathcal{U}_i^t} \sum_{x_i \in \mathcal{X}_i} \alpha_i^t(u_i^t, x_i; I_i^t, Z_i^t) p(Y_i|x_i, I_i^t; \mu_i^{1:t-1})$$

with

$$\alpha_i^t(u_i^t, x_i; I_i^t, Z_i^t) = p(x_i) P_i^t(I_i^t, Z_i^t|x_i) \left[c(u_i^t, x_i) + C_i^t(I_i^{t+1}, x_i)\right].$$

Remark: Arguably the most optimistic part of Conjecture 5.1 is the lack of explicit dependence on Y_i in the cost function C_i^t. With such dependence, the optimal communication rule μ_i^t would not necessarily lie in a finitely-parameterized subset of $\mathcal{M}_i^t$. In turn, the connection between iterating the associated fixed-point equations and executing an *exact* coordinate-descent algorithm over the original function space Γ, which is how the convergence guarantees in earlier chapters were deduced, would be lost.

■ 5.4 An Approximate Offline Algorithm

The analysis of the preceding section reveals how generalizing to a multi-stage online processing model brings forth a number of new barriers to tractably computing team-optimal decision strategies. On the positive side, Proposition 5.2 establishes the minimal assumptions under which online computation scales linearly in the number of nodes n. These assumptions, namely conditional independence and measurement/channel locality, are seen to coincide with those needed to guarantee online efficiency (of the final-stage detection strategy δ^*) in the single-stage architectures. However, in contrast to the single-stage cases, Proposition 5.3 establishes that then adding the cost locality assumption is *not* enough to guarantee that the associated offline computation scales linearly in n. Moreover, we were unable to derive analogous structural results for the multi-stage communication strategy μ^*, offering instead Conjecture 5.1 that proposes it enjoys the analogous online efficiency of its single-stage counterparts. Indeed, we expect the offline computation associated with μ^* to be no easier than that of the final-stage detection strategy δ^*, considering the latter need only account for the receivers' perspectives of any multi-stage signaling incentives whereas the former should also account for the transmitters' perspectives.

Supposing Assumptions 5.1–5.4 are in effect, this section describes an approximate offline algorithm for generating multi-stage measurement processing strategies. We stress that this approximation is only suited for a small number of online stages T, as

it continues to respect the parameterization suggested by Conjecture 5.1 and Proposition 5.3 and, hence, assumes the exponential growth in T is not yet a barrier. In this light, the approximation is most useful for addressing what performance benefits are achievable when moving from single-stage to two-stage architectures, from two-stage to three-stage architectures and so on as long as T is small enough such that local memory requirements remain manageable. Of course, of equal interest is the question of finding good limited-memory approximations for problems that merit large T, a pursuit we will have to leave for future research.

■ 5.4.1 Overview and Intuition

Before describing our approximation in full detail, let us develop an intuitive understanding of the overall procedure at a high level. There are two main steps:

1. find a particular communication strategy $\tilde{\mu} \in \mathcal{M} = \mathcal{M}_1 \times \cdots \times \mathcal{M}_n$ by making iterative use of the efficient single-stage algorithms derived in preceding chapters;

2. find a particular detection strategy $\tilde{\delta} \in \Delta = \Delta_1 \times \cdots \times \Delta_n$ by applying Proposition 5.3 and, for problems in which the computation in (5.10) becomes impractical, employing a sampling-based approximation to obtain the required statistics $P_i^{\tilde{\mu}}$ for each node i.

Recall that Proposition 5.3 characterizes the optimal detection strategy, assuming the multi-stage communication strategy is fixed, so approximations are made primarily within the procedure by which we first generate the communication strategy $\tilde{\mu}$.

Our approximation of the stage-one communication rules $\tilde{\mu}^1 = (\tilde{\mu}_1^1, \ldots \tilde{\mu}_n^1)$ is straightforward, as no node has yet to account for local memory. Applying the single-stage solution (i.e., assuming the final decisions are made right after this single stage communication), the obtained communication rule $\tilde{\mu}_i^1$ for every node i is a member of the stage-one function space $\mathcal{M}_i^1$. Of course, the single-stage approximation fails to capture incentives for impacting the value of later-stage transmissions. Indeed, this side of the multi-stage signaling incentives is neglected throughout our approximation, as we repeatedly use the single-stage solutions without any look-ahead to future rounds of communication.

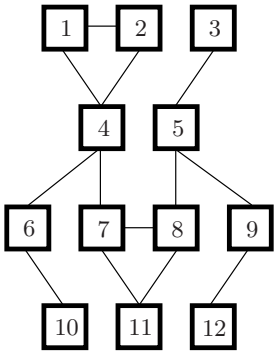
The rules $\tilde{\mu}^{2:T} = (\tilde{\mu}_1^{2:T}, \ldots, \tilde{\mu}_n^{2:T})$ for all subsequent stages are selected in parallel for each node i. Doing so clearly neglects the fact that the true conditional distribution $p(I_i^t, z_i^t | x_i; \tilde{\mu}^{1:t-1})$ for every node-stage pair (i,t) is a function of all nodes' communication rules $\tilde{\mu}^{1:t-1}$ from previous stages. Specifically, computing each such conditional

distribution involves an analogously global computation as that described for the statistics $P_i^{\tilde{\mu}}(u_i, z_i|x_i) = p(I_i^T, z_i^{T+1}|x_i; \tilde{\mu})$ in the final-stage detection rule local to node i. We avoid having to keep track of all nodes' communication rules from previous stages by essentially assuming that, when constructing the single-stage problems at node i and stage t to determine the communication rule $\tilde{\mu}_i^t$, the side information Z_i^t is *unrelated* to local memory I_i^t. We do, however, properly account for the local memory I_i^t inside of the measurement likelihood $p(y_i|x_i, I_i^t; \tilde{\mu}^{1:t-1})$ in accordance with Lemma 5.2. Moreover, all of the other nodes still appear within the single-stage problems for node i, but we extract only the rule local to node i from each single-stage solution for use in the actual communication rule $\tilde{\mu}_i^t$. Finally, the manner in which we construct the series of single-stage problems for each node i, including how we craft its local models from the given multi-stage models, involves other significant yet subtle approximations. These are described in detail in the following subsections, but the main ideas are illustrated in Figure 5.3 by way of an example.

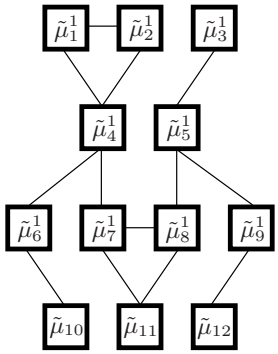
In summary, there are three main sources of approximation in the procedure we use to construct a feasible multi-stage communication strategy $\tilde{\mu} \in \mathcal{M}$. Firstly, we do not know whether the finite parameterization proposed by Conjecture 5.1 is correct. Secondly, the stage-t communication rule of every node i is designed assuming it is the final round of communication. Thirdly, the side information Z_i^t is represented in each single-stage approximation as the output of phantom nodes, neglecting its true dependence on all nodes' communication rules $\tilde{\mu}^{1:t-1}$ in the preceding stages. The overall approximation, however, does preserve two important structural attributes of the optimal multi-stage strategy. The first is the memory-dependent paring down of all nodes' local likelihoods, in accordance with Lemma 5.2. The second is that, for the particular selected communication strategy $\tilde{\mu}$, we employ the team-optimal final-stage detection strategy given by Proposition 5.3. These two attributes are key to preserving satisfactory performance of the multi-stage strategy $\tilde{\gamma} = (\tilde{\mu}, \tilde{\delta})$ despite the approximations made to select the communication strategy $\tilde{\mu}$.

■ 5.4.2 Step One: Approximating the Communication Strategy

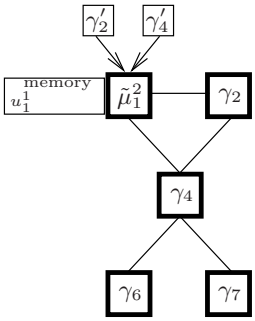
Our method for constructing an approximate multi-stage communication strategy $\tilde{\mu} \in \mathcal{M}$ combines the probabilistic structure exposed by Lemma 5.2 and Lemma 5.3, the finite parameterization proposed by Conjecture 5.1, and repeated application of the single-stage offline message-passing algorithms derived in previous chapters. The outer loop of the algorithm proceeds over increasing stages, followed by an inner loop over



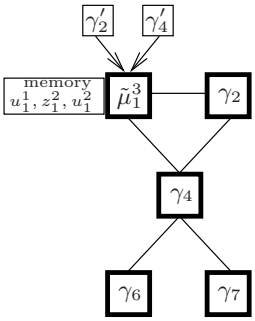
(a) An Original Undirected Network



(b) Single-Stage Network for $t = 1$



(c) Single-Stage Network for $(i, t) = (1, 2)$



(c) Single-Stage Network for $(i, t) = (1, 3)$

Figure 5.3. (a) A specific undirected network topology $\mathcal{F}$ in a multi-stage problem and (b)-(d) the sequence of single-stage hybrid topologies constructed from the perspective of node $i = 1$. All first-stage rules (including that of node i) can be approximated by just one single-stage solution, whereas the advent of memory in subsequent communication stages requires a single-stage solution per value of the local memory I_i^t. Rules γ_j for $j \neq i$ represent functions that are optimized within every single-stage solution, but then discarded once the others are selected for the multi-stage strategy. Note the introduction of phantom non-leader nodes to account for the presence of side information Z_i^t local to node i, which in the multi-stage strategy results from its neighbors' decisions $U_{ne(i)}^{t-1}$ but in the single-stage approximation is simply optimized from scratch. Moreover, because we extract only the communication rule local to node i for use in our multi-stage strategy, we need not include the nodes that lie beyond its two-step neighborhood in $\mathcal{F}$.

nodes, constructing firstly the nodes' stage-one communication rules $\tilde{\mu}^1 \in \mathcal{M}^1 = \mathcal{M}_1^1 \times \cdots \times \mathcal{M}_n^1$, secondly the nodes' stage-two communication rules $\tilde{\mu}^2 \in \mathcal{M}^2$ holding $\tilde{\mu}^1$ fixed, and so on through the nodes' stage-T communication rules $\tilde{\mu}^T \in \mathcal{M}^T$ holding $\tilde{\mu}^1, \tilde{\mu}^2, \ldots, \tilde{\mu}^{T-1}$ fixed. For each particular stage t and node i, there is an inner-most loop over all values of local memory I_i^t, crafting a series of single-stage problems whose solutions (via the efficient offline message-passing algorithms of the preceding chapters)

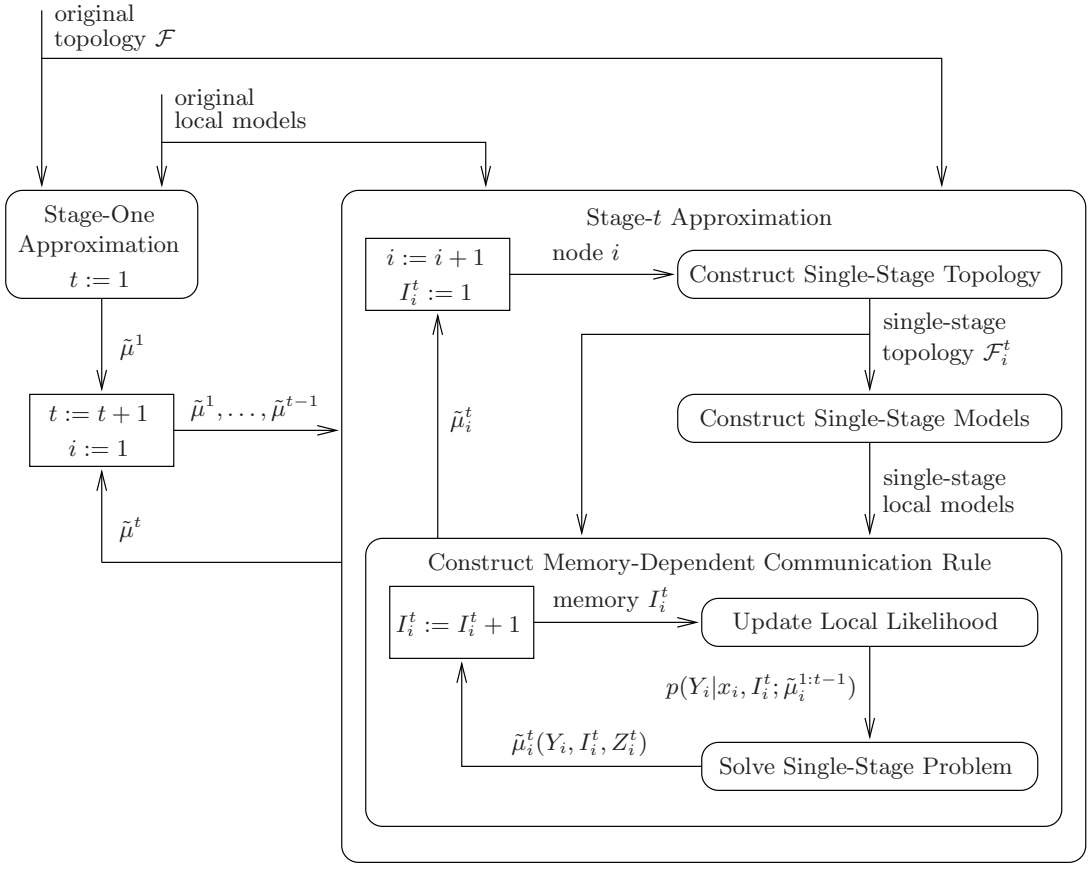


Figure 5.4. A high-level flowchart of our algorithm for constructing an approximate multi-stage communication strategy $\tilde{\mu}$. In stage $t = 1$, every node's information vector is empty and the single-stage approximation operates directly on the given network topology $\mathcal{F}$, yielding all nodes' initial communication rules $\tilde{\mu}^1$. In successive stages $t > 1$, there is an inner loop over all nodes and, for each node i, an inner-most loop over all possible values of local memory I_i^t, crafting a series of single-stage problems whose solutions collectively determine all nodes' stage-t communication rules $\tilde{\mu}^t$.

collectively determine a particular local communication rule $\tilde{\mu}_i^t \in \mathcal{M}_i^t$. A high-level flowchart of this algorithm is shown in Figure 5.4, and the remainder of this subsection describes the details related to each module.

Constructing Single-Stage Network Topologies

Consider any stage $t > 1$ and any particular node i. We determine the stage-t communication rule local to node i via a series of single-stage approximations, all relying on a particular topology we denote by $\mathcal{F}_i^t$ to be constructed in a manner that depends on

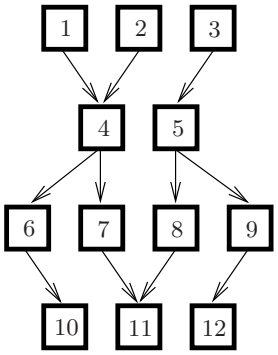
whether the original network topology $\mathcal{F}$ is undirected or directed. In either case, the objective is to preserve a compatible structural form between the (memory-dependent) multi-stage communication rules proposed in Conjecture 5.1 and the collection of communication rules resulting from this series of single-stage approximations. Details for constructing each single-stage network $\mathcal{F}_i^t$ to achieve this objective are as follows.

Let us first describe the case given the original network topology $\mathcal{F}$ is undirected, defining for every node i the neighbors $ne(i)$ and the two-step neighborhood $ne^2(i) = \cup_{j \in ne(i)} ne(j) - \{i\}$. The network $\mathcal{F}_i^t$ in the single-stage approximation is taken to be the hybrid network in which the (undirected) leader network is the subgraph of $\mathcal{F}$ induced by nodes $i \cup ne^2(i)$, while the (directed) non-leader network duplicates the nodes in $ne(i)$ and assigns each a single outgoing link to leader node i. These phantom nodes act as surrogates for representing the side information Z_i^t based on the decisions $U_{ne(i)}^{t-1}$ from the preceding communication stage. Figure 5.3 shows a particular undirected network $\mathcal{F}$ and illustrates these single-stage networks for a specific node i. Notice that the sets $\mathcal{Z}_i^D$ and $\mathcal{U}_i^U$ in the hierarchical fusion architecture given $\mathcal{F}_i^t$ are identical to the sets $\mathcal{Z}_i^t$ and $\mathcal{U}_i^t$, respectively, in the multi-stage architecture given $\mathcal{F}$.

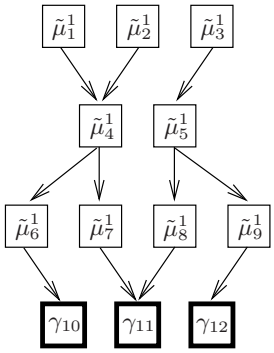
We now describe the case given original network topology $\mathcal{F}$ is directed, defining the parents $pa(i)$ and children $ch(i)$ for each node i. Recall from Figure 5.1 that when stage $t > 1$ is odd (even), the flow of information proceeds in the forward (backward) partial order implied by $\mathcal{F}$. Accordingly, the construction of $\mathcal{F}_i^t$ for each stage-node pair similarly depends on whether t is odd or even, as well as whether node i is a pivot node (i.e., a parentless or childless node in $\mathcal{F}$ for t odd or even, respectively). Suppose $t > 1$ is odd: unless node i is parentless, it has the same parents and children in $\mathcal{F}_i^t$ as it has in $\mathcal{F}$; however, if node i is parentless, we duplicate the children of node i in $\mathcal{F}$ and designate them as phantom parents of node i in $\mathcal{F}_i^t$. Similarly suppose t is even: unless node i is childless, its parents and children in $\mathcal{F}$ become its children and parents, respectively, in $\mathcal{F}_i^t$; however, if node i is childless, we duplicate the parents of node i in $\mathcal{F}$ and designate them as the phantom parents of node i in $\mathcal{F}_i^t$. Figure 5.5 shows a particular directed network $\mathcal{F}$ and illustrates these single-stage networks for a specific node i. Notice that the sets $\mathcal{Z}_i$ and $\mathcal{U}_i$ given the single-sweep network $\mathcal{F}_i^t$ are identical to the sets $\mathcal{Z}_i^t$ and $\mathcal{U}_i^t$, respectively, in the multi-stage architecture given $\mathcal{F}$.

Constructing Single-Stage Local Models

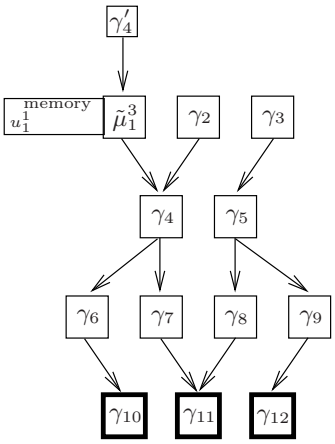
At this point in our algorithm, we are given (i) a particular stage-node pair (t, i) and (ii) the topology $\mathcal{F}_i^t$ constructed from the original network topology $\mathcal{F}$ as just described.



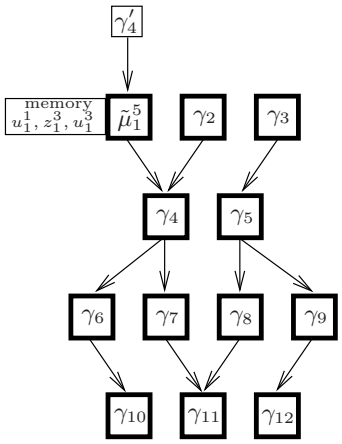
(a) An Original Directed Network



(b) Single-Stage Network for $t = 1$



(c) Single-Stage Network for $(i, t) = (1, 3)$



(c) Single-Stage Network for $(i, T) = (1, 5)$

Figure 5.5. (a) A specific directed network topology $\mathcal{F}$ in a multi-stage problem with $T = 5$ and (b)-(d) the sequence of single-stage topologies constructed from the perspective of a pivot node i. Being parentless in $\mathcal{F}$, node i generates a communication decision on only odd-numbered stages. All first-stage rules (including that of node i) can be approximated by just one single-stage solution, whereas the advent of memory in subsequent communication stages requires a single-stage solution per value of the local memory I_i^t. Rules γ_j for $j \neq i$ represent functions that are optimized within every single-stage solution, but then discarded once the others are selected for the multi-stage strategy. Note the introduction of phantom parents to account for the presence of side information Z_i^t local to node i, which in the multi-stage strategy results from its children's decisions $U_{ch(i)}^{t-1}$ in the preceding even-numbered stage but in the single-stage approximation is simply optimized from scratch. All but the final stage $T = 5$ assume only the childless nodes in $\mathcal{F}$ are in the gateway, consistent with us seeking communication-only rules during stages $t < T$.

The task is to select a local stage-t communication rule $\tilde{\mu}_i^t \in \mathcal{M}_i^t$. This is accomplished by applying the offline message-passing algorithms of previous chapters to a series of single-stage problems defined on the network $\mathcal{F}_i^t$, using a collection of local models

crafted for each stage-node pair as follows.

Let us first specify the local cost models in each single-stage problem based on the network topology $\mathcal{F}_i^t$. Recall that the original local cost models collectively satisfy Assumption 5.4, where parameter λ is given and all nodes in $\mathcal{F}$ are assumed to be in the gateway. In the single-stage approximation, however, the goal is merely to approximate the stage-t communication rule local to node i, so we select its gateway nodes accordingly. In particular, if $\mathcal{F}$ is undirected, then the single-stage gateway consists only of the leader nodes in $\mathcal{F}_i^t$, while if $\mathcal{F}$ is directed, then the gateway consists only of the childless nodes in $\mathcal{F}_i^t$. In either case, each such gateway node j in $\mathcal{F}_i^t$ uses the original detection-related costs $c(\hat{x}_j, x_j)$. We must also specify communication-related costs for every node with at least one child in $\mathcal{F}_i^t$: if this node is a phantom of some node j in $\mathcal{F}$, then we use the previous-stage costs $c(u_j^{t-1}, x_j)$, whereas if this node is an actual node j in $\mathcal{F}$, then we use the current-stage costs $c(u_j^t, x_j)$.

We next specify the local measurement/channel models in each single-stage problem based on the network topology $\mathcal{F}_i^t$. Recall that the original measurement/channel models collectively satisfy Assumption 5.2. Every node in $\mathcal{F}_i^t$ is either an actual node j in $\mathcal{F}$ or a phantom of some node j in $\mathcal{F}$: in either case, we use the original measurement model $p(y_j|x_j)$. Because each phantom is parentless in $\mathcal{F}_i^t$, local channel models are needed only for nodes in $\mathcal{F}_i^t$ that correspond to actual nodes in $\mathcal{F}$. Recall that the original channel models satisfy Assumption 5.1. When $\mathcal{F}_i^t$ is a hybrid network, only node i has neighbors in both the leader and non-leader network, so we use $p(z_i^t|x_i, u_{tr(i,t)})$ to describe the symbol(s) received from the former and $p(z_i^{t-1}|x_i, u_{tr(i,t-1)})$ to describe the symbol(s) received from the latter. Every other leader node $j \neq i$ has only neighboring leaders, so we use $p(z_j^t|x_j, u_{tr(j,t)})$ to describe its received symbol(s). When $\mathcal{F}_i^t$ is a directed network, every non-phantom j with at least one parent and at least one child in $\mathcal{F}_i^t$ uses $p(z_j^t|x_j, u_{tr(j,t)})$ to describe its received symbol(s); however, recall from Figure 5.1 that node j is childless in $\mathcal{F}_i^t$ only if it corresponds to a pivot node for the next stage $t+1$, so by convention $p(z_j^t|x_j, u_{tr(j,t)}) = 1$ and we instead use $p(z_j^{t+1}|x_j, u_{tr(j,t+1)})$ to describe its received symbol(s).

It remains to specify the local prior models in each single-stage problem based on the network topology $\mathcal{F}_i^t$. Recall our convention that $p(x_j, x_{ne(j)})$ is known for every node j in the original network topology $\mathcal{F}$. Observe that a phantom node is always parentless in $\mathcal{F}_i^t$, so its local prior model is simply the marginal $p(x_j)$ of the corresponding actual node j in $\mathcal{F}$. Next consider any non-phantom j in $\mathcal{F}_i^t$, including the specific node i. When $\mathcal{F}_i^t$ is a hybrid network, every such j is a leader node, so $p(x_j, x_{ne(j)})$ characterizes

its neighborhood prior for the leader network; node i also requires a local prior involving its parents $\tilde{pa}(i)$ in the non-leader network, which are always phantoms of $ne(i)$ and thus we use $p(x_i, x_{\tilde{pa}(i)}) = p(x_i, x_{ne(i)})$. When $\mathcal{F}_i^t$ is a directed network and the stage t is odd (even), the parents $\tilde{pa}(j)$ of each non-phantom j in $\mathcal{F}_i^t$ are exactly the parents (children) of node j in $\mathcal{F}$, so we use

$$p(x_j, x_{\tilde{pa}(j)}) = \begin{cases} \sum_{x_{ch(j)}} p(x_j, x_{ne(j)}) & , \text{ if stage } t \text{ odd} \\ \sum_{x_{pa(j)}} p(x_j, x_{ne(j)}) & , \text{ if stage } t \text{ even} \end{cases} .$$

Constructing Memory-Dependent Communication Rules

At this point in our algorithm, the network topology $\mathcal{F}_i^t$ and the associated local models completely specify a single-stage problem amenable to the efficient message-passing algorithms of preceding chapters. However, for its solution to yield a communication rule for node i that is compatible with the stage-t communication rule proposed in Conjecture 5.1, the single-stage problem must also account for the local memory I_i^t. This is accomplished by looping over all values of local memory I_i^t, in each instance using the pared-down likelihood function $p(y_i|x_i, I_i^t; \tilde{\mu}_i^{1:t-1})$ defined via (5.7) as the measurement model local to node i. This ensures that the associated single-stage solution yields a communication rule for node i that consists of up to $|\mathcal{Z}_i^t|$ distinct size-$|\mathcal{U}_i^t|$ partitions of the restricted measurement space $\mathcal{Y}_i(I_i^t; \tilde{\mu}_i^{1:t-1})$. Such a communication rule, by virtue of Lemma 5.2, coincides with the stage-t communication rule local to node i for that fixed value of memory I_i^t. In turn, by repeating this procedure over all values of local memory I_i^t, the series of single-stage rules collectively defines a memory-dependent communication rule $\tilde{\mu}_i^t$ lying in the finitely-parameterized subspace of $\mathcal{M}_i^t$ proposed by Conjecture 5.1.

■ 5.4.3 Step Two: Approximating the Detection Strategy

Given Assumptions 5.1–5.4 are satisfied and the multi-stage communication strategy is fixed to some member of the function space $\mathcal{M}$, direct application of Proposition 5.3 is guaranteed to minimize the detection penalty over the function space Δ. In other words, to find the best (online) detection strategy $\tilde{\delta}$ for the approximate multi-stage communication strategy $\tilde{\mu}$, it suffices to compute (offline) the likelihood function $P_i^{\tilde{\mu}}(u_i, z_i|x_i) \equiv P_i^{\tilde{\mu}}(I_i^{T+1}, z_i^{T+1}|x_i)$ for every node i. Of course, as was emphasized in Section 5.3, exact computation of these likelihood functions appears to be intractable for even modestly-sized networks. While (5.10) and (5.13) reveal that computing these

likelihood functions involve taking sums over distributions that exhibit a factored form, the development of methods to exploit this special structure is left for future work.

In our preliminary experiments with the multi-stage architectures, described in the next section, we rely on simulation-based approximations to the desired likelihood functions $P_i^{\tilde{\mu}}$. Specifically, we draw independent samples from the joint distribution $p(x, y)$, and for each such sample apply both the multi-stage communication strategy $\tilde{\mu}$ and sample from the local channel models to yield a specific sequence of transmitted/received symbols (u_i, z_i) local to each node i. We then approximate $P_i^{\tilde{\mu}}$ with the empirical conditional distribution calculated from these generated samples i.e., if $N(x_i, u_i, z_i)$ denotes the number of samples in which node i realizes the triplet (x_i, u_i, z_i), we employ

$$P_i^{\tilde{\mu}}(u_i, z_i|x_i) \approx \begin{cases} \dfrac{N(x_i, u_i, z_i)}{\sum_{u_i, z_i} N(x_i, u_i, z_i)} & , \text{ if } \sum_{u_i, z_i} N(x_i, u_i, z_i) > 0 \\ 0 & , \text{ otherwise} \end{cases} .$$

A practical caveat of this empirical approximation is worth mentioning. Firstly, note that certain pairs (u_i, z_i) of symbols visible to node i may have zero probability of occurrence under the fixed strategy $\tilde{\mu}$. However, identifying *a priori* all such improbable pairs (u_i, z_i) can be challenging. Moreover, with only a finite number of samples, it is possible that probable triplets (x_i, u_i, z_i) are never actually generated. Thus, we must handle zeros in the empirical distribution with additional care. In particular, if $N(x_i, u_i, z_i)$ is zero but the sum $\sum_{x_i} N(x_i, u_i, z_i)$ is nonzero, then we reassign $P_i^{\tilde{\mu}}(u_i, z_i|x_i)$ to its smallest value over all empirically probable events i.e.,

For every (x_i, u_i, z_i) such that $N(x_i, u_i, z_i) = 0$ but $\sum_{x_i} N(x_i, u_i, z_i) > 0$, reassign

$$P_i^{\tilde{\mu}}(u_i, z_i|x_i) := \min_{\{(u_i', z_i')|P_i^{\tilde{\mu}}(u_i', z_i'|x_i)>0\}} P_i^{\tilde{\mu}}(u_i', z_i'|x_i).$$

This adjustment recognizes that, assuming every measurement $y_i \in \mathcal{Y}_i$ is probable no matter the value of the (hidden) state X_i, the pair (u_i, z_i) must be probable for every $x_i \in \mathcal{X}_i$ if it is probable for at least one $x_i \in \mathcal{X}_i$. This same caveat carries over to online processing: that is, it is possible that no instance of a probable pair (u_i, z_i) is ever observed during the offline sampling procedure. Thus, if the online rule encounters a pair (u_i, z_i) that was deemed improbable by offline computation (i.e., the sum $\sum_{x_i} N(x_i, u_i, z_i)$ was zero), the above adjustment to $P_i^{\tilde{\mu}}$ is made for *every* $x_i \in \mathcal{X}_i$ before proceeding with the final-stage decision $\hat{x}_i = \tilde{\delta}_i(y_i, u_i, z_i)$.

■ 5.5 Examples and Experiments

This section summarizes experiments with the multi-stage decision architectures and the approximate offline algorithm just described. Throughout, the global sensing, communication and cost models are essentially the same as those employed in the experiments of previous chapters, altogether depending on just four parameters w, r, q and λ. In particular, the hidden state process X consists of n spatially-distributed binary random variables, their pairwise interactions defined by an undirected graphical model with common parameter $w \in [0, 1]$ for all edge potentials as was first described in Subsection 3.4.2. The global measurement process Y consists of n identical linear Gaussian detectors, their spatially-independent noise processes parameterized by a common value of $r \in (0, \infty)$ as was first described in Subsection 3.4.1. The multi-stage communication model is taken to be a selective broadcast transmission scheme (see Example 3.3) along with stationary & memoryless interference channels, meaning every node's local channel model in every stage depends on a common unreliability parameter $q \in [0, 1]$ as was described for single-stage architectures in Subsection 4.5.4. Finally, as was first described in Subsection 3.4.1, the global costs are chosen to optimize the sum of the gateway node-error-rate J_d and network-wide link-use-rate J_c (weighted by λ). The only difference in the multi-stage case is that the latter measures the sum over *all* communication stages (i.e., the maximum value of J_c is T times that of the single-stage counterpart).

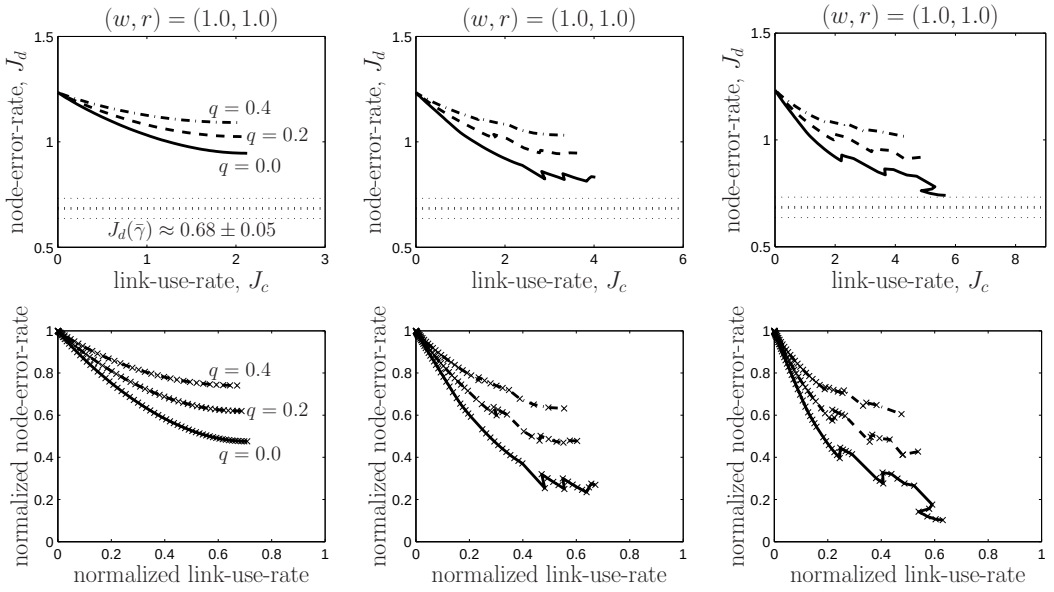
The scope of these experiments is to investigate the extent to which global decision-making performance can improve when we generalize to multiple online communication stages. To this end, our initial focus is on a hidden Markov model (i.e., see Example 2.9), for which our decision problem is easily solved in the absence of explicit network constraints. Our results for a four-node instance of this model show that, in either a directed or undirected multi-stage architecture, decentralized detection performance approaches that of the optimal (i.e., centralized, or unconstrained) strategy in as little as $T = 3$ decision stages. We then move to a four-node "loopy" graphical model, instances of which belong to the class of so-called "frustrated" models known to present difficulty for most existing local message-passing approximations. Experiments on this loopy model paint a number of different algorithmic comparisons between our team-theoretic approximations and those inspired by belief propagation algorithms. Altogether, while our methods are superior from the communication overhead perspective (by design), comparisons from the computation overhead or decision performance perspectives are less clear cut. That is, the relative advantages and disadvantages appear

to be application-dependent and even model-dependent, raising a host of new questions for future research to be discussed in Chapter 6.

■ 5.5.1 A Small Hidden Markov Model

An n-node hidden Markov model (Example 2.9) is arguably the most commonly studied probabilistic graphical model. Each node i in the underlying graph $\mathcal{G}$ has (at most) two neighbors. Computing the posterior marginal $p(x_i|y)$ at every node i, which under our minimum node-error-rate criterion is a sufficient statistic for deciding the optimal value of component state estimate $\hat{x}_i$, is straightforward via (unconstrained) belief propagation methods. The Viterbi algorithm, starting from a directed graphical representation for X, yields the correct marginals after only a single forward-backward sweep on the probability graph; the sum-product algorithm, starting from an undirected graphical representation for X, is guaranteed to converge with the correct marginals after n parallel message-passing iterations on the probability graph. The experiments we now describe focus on the simple case of only $n = 4$ nodes, but imposing explicit online processing constraints that render the standard belief propagation algorithms infeasible. To be specific, instead of assuming the reliable (online) communication of real-valued messages, we restrict the messages to ternary-valued symbols (i.e., each link is unit-capacity with each node given a "no-send" option) and each link can be unreliable (i.e., a binary-valued symbol actually transmitted is not always successfully received).

The first main question we address empirically here is whether the multi-stage approximation we described in Section 5.4 has *any* hope of adequately capturing the sophisticated performance/communication tradeoffs demonstrated in previous chapters. Recall how, for the single-stage architectures, the family of strategies obtained by our offline algorithms (i.e., over a range of values for $\lambda \geq 0$) monotonically trades decreasing detection penalty with increasing communication penalty. Moreover, the tradeoff became less pronounced as network reliability degraded, all other things equal e.g., larger values for erasure probability q yielded smaller reductions in node-error-rate per unit increase in link-use-rate. Here, we set parameters $w = r = 1$ and consider three different degrees of network reliability, namely $q = 0$, $q = 0.2$ and $q = 0.4$. Recall that setting w to unity corresponds to a global binary state model (i.e., all four hidden states are equal with probability one). Considering this special case keeps the problem small enough to permit direct computation of the final-stage detection strategy for up to $T = 3$ communication stages. Avoiding the sampling-based approximation of the final-stage detection strategy controls our experiments in two useful ways: firstly, we

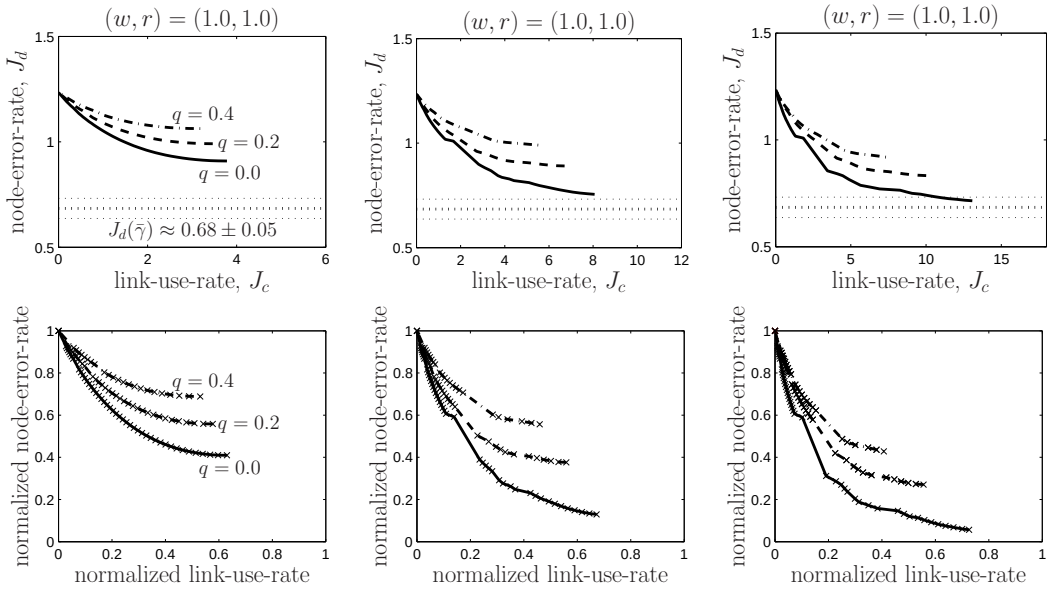


(a) $T = 1$ Stage (b) $T = 2$ Stages (c) $T = 3$ Stages

Figure 5.6. Optimized tradeoff curves achieved by our team-theoretic approximations given the four-node directed tandem network described in Subsection 5.5.1 assuming (a) $T = 1$, (b) $T = 2$ and (c) $T = 3$ online communication stages. The efficient message-passing solution of Chapter 3 is directly applicable to the single-stage architecture in (a), and repeatedly applied within the approximate offline algorithm described in Section 5.4 for the multi-stage architectures in (b) and (c). Each curve is obtained by varying λ from zero (in increments of 0.005) up to the first value in which the myopic strategy becomes optimal. Also shown is a Monte-Carlo estimate of the optimal centralized performance $J_d(\bar{\gamma})$, using 1000 samples. The second row of figures uses the same data as in the first, normalizing the two penalties to better compare across the different number of stages. The $\times$'s in this second row of figures mark the specific points $(J_c^\lambda, J_d^\lambda)$ associated with the chosen λ values $0, 0.005, 0.010, \ldots$, and each curve connects these points with line segments in the order of decreasing λ. Note the non-uniformity of the $\times$'s in (b) and (c) as compared to (a), clearly an artifact of the multi-stage approximation in comparison to the guaranteed team-optimality in (a). See Subsection 5.5.1 for more discussion of these results.

can compute the multi-stage performance $(J_c^\lambda, J_d^\lambda)$ for each fixed value of λ exactly, without relying on Monte-Carlo estimates; and secondly, we can attribute any suspect results entirely to our approximation of the multi-stage communication strategy.

Figure 5.6 displays the resulting collection of tradeoff curves given a directed network topology (i.e., the four node tandem topology) over nine different values of parameters (q, T). Indeed, we see the same general dependence on parameter q for the multi-stage architectures as we observed in the single-stage architectures. However, inspecting the tradeoff curves for each individual value of q more closely, the observed



(a) $T = 1$ Stage (b) $T = 2$ Stages (c) $T = 3$ Stages

Figure 5.7. The analogous results as presented in Figure 5.6, except considering the *undirected* network constraints described in Subsection 5.5.1. Note that the second row of figures still show non-uniformity of the $\times$'s in (b) and (c); however, in contrast to the curves shown in Figure 5.6 for the directed network constraints, the multi-stage curves here continue to exhibit a monotonic tradeoff between increasing communication penalty and decreasing detection penalty. See Subsection 5.5.1 for more discussion of these results.

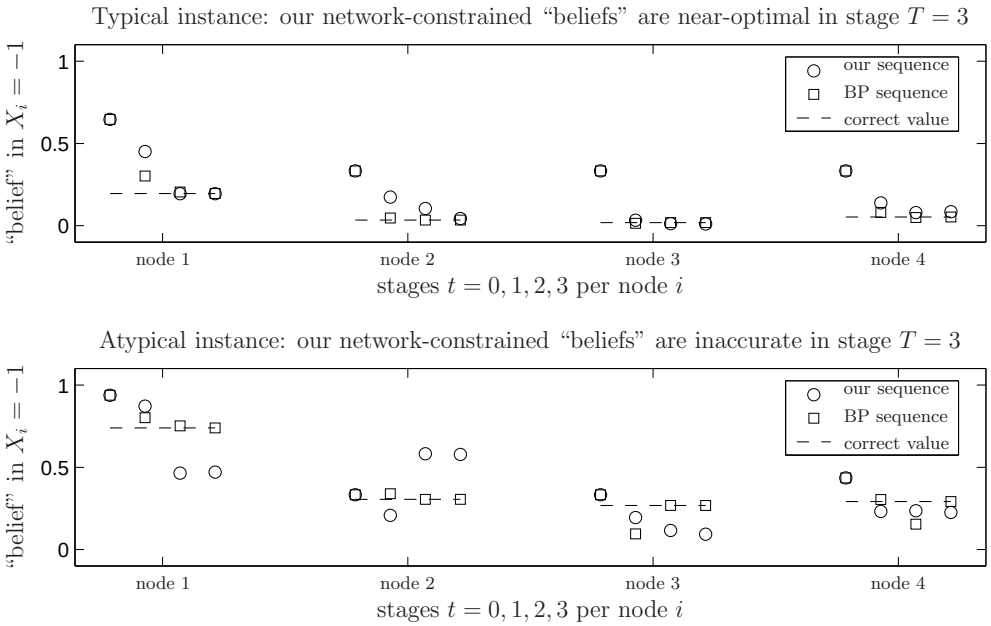
non-monotonicity when $T > 1$ implies that our multi-stage approximation does not *always* yield improved detection performance upon tolerating additional communication penalty. Nonetheless, on the whole, these tradeoff curves do resemble those of the single-stage architecture; moreover, we see that the achieved detection performance gets significantly closer to the benchmark centralized performance with each additional communication stage, all other things equal.

Figure 5.7 displays the analogous collection of tradeoff curves given an undirected network topology, every stage of communication featuring bidirectional symbol exchanges along each link. Here, just as in the case of directed constraints, on the whole the multi-stage approximation achieves the same type of performance tradeoffs as those achieved by the team-optimal single-stage solution. Interestingly, in comparison to Figure 5.6, the non-monotonicity of each individual curve is not nearly as apparent. This suggests that the multi-stage approximation may somehow be better tuned for the probabilistic structure induced by undirected network constraints than those induced

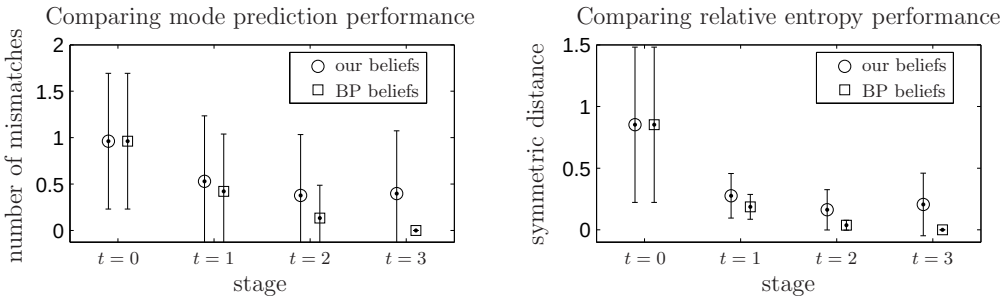
by directed network constraints.

The second main question we address empirically is how well *network-constrained* posterior marginals, generated via (5.12) in successive stages of our multi-stage decentralized strategy, can approximate those generated by successive iterations of the (unconstrained) belief propagation algorithm. In a four-node hidden Markov model, the belief propagation algorithm is known to converge in just three parallel iterations, always yielding the exact posterior marginal $p(x_i|y)$ at every node i. We focus on the instance of this model with $(w, r) = (0.9, 1)$. The analogous network-constrained architecture is that of three-stages with the same undirected topology as the probability graph, setting both the erasure probability q and the weight λ on communication penalty to zero to most closely match the ideal communication assumptions of belief propagation. Note that with these parameter settings, we may view a multi-stage communication strategy as a (severely) quantized analog to the belief propagation algorithm, every pair of neighboring nodes successively exchanging ternary-valued symbols as opposed to real-valued messages. Of course, success of our network-constrained solution distinctly requires the initial investment in offline optimization. In particular, to implement (5.12), we must both select a sound communication strategy $\tilde{\mu}$ and determine the associated final-stage likelihood function $P_i^{\tilde{\mu}}$ for every node i. In the experiments to follow, these quantities were found using the approximate offline algorithm described in Section 5.4, generating the former via repeated applications of the single-stage algorithm and the latter based on 10000 samples from the processes (X, Y) and simulating the processes $(U, Z) = \tilde{\mu}(Y)$.

Figure 5.8 compares the sequence of network-constrained "beliefs" (i.e., approximation of the true posterior marginals) given by our multi-stage decentralized strategy to those given by the (unconstrained) belief propagation algorithm. Figure 5.8(a) shows two instances of these belief sequences, the only difference between the two being the measurement vector $Y = y$. The first instance shows the network-constrained approximation being close to belief propagation after $T = 3$ iterations, while the second instance exhibits disagreement between the two. The latter case turns out to be an atypical case, as is reflected by the on-average performance comparison in Figure 5.8(b). These are calculated based on 1000 samples from the measurement process Y, applying two different measures of error on each resulting sequence of beliefs. The first we call the mode-prediction error, which quantifies how often the mode of a node's "belief" differs from the mode of its true posterior marginal. Specifically, denoting the stage-t belief at node i by M_i^t, we count a mismatch at node i in stage t if $\arg\max_{x_i} p(x_i|y) \neq \arg\max_{x_i} M_i^t(x_i)$.



(a) Successive "beliefs" generated by our strategy versus those by belief propagation



(b) Average performance of our network-constrained "beliefs" versus those of belief propagation

Figure 5.8. Empirical comparison between the sequence of "beliefs" (i.e., approximation of the true posterior marginals) produced by our network-constrained strategy and those produced by (unconstrained) belief propagation in a four-node hidden Markov model. In (a), we show the specific belief sequences for two of the 1000 samples from measurement process Y, while (b) compares their on-average performance per stage t. Belief propagation always converges to the correct answers in $T = 3$ stages, and our network-constrained approximation is seen to remain within statistical significance over successive stages under two different error measures.

The mode prediction error per stage sums these mismatches over the four nodes, then taking the average over 1000 measurement samples. The second measure of error is the (symmetrized) relative entropy [22] between $M_i^t(x_i)$ and $p(x_i|y)$, again taking the sum

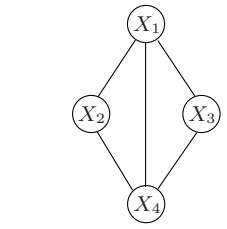
over nodes for each sample and then averaging the result over all 1000 samples. Under either error measure, we see that our network-constrained beliefs stay within statistical significance of the errors associated with the (optimal) belief propagation algorithm.

■ 5.5.2 A Small "Loopy" Graphical Model

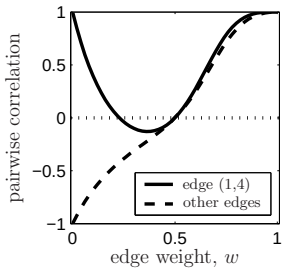
It is well-known that graphical models with cycles, or loops, present many additional computational challenges in comparison to their tree-structured counterparts. Indeed, most iterative message-passing algorithms such as belief propagation are derived assuming the absence of loops, so their application to such models raises deep questions about convergence and, given convergence does occur, the quality of the resulting solution. A simplest example that exposes the associated limitations of loopy belief propagation is the four-node model shown in Figure 5.9(a). We see in Figure 5.9(c) that convergence (and hence satisfactory decision-making performance) of loopy belief propagation is lost for parameter values of w near zero, corresponding to models in which every pairwise interaction is (locally) repulsive. This can be attributed to the net effect of the pair of cycles, shown in Figure 5.9(b) to make the central edge between X_1 and X_4 become attractive as w tends to zero. Efficient message-passing algorithms are known to have difficulty when "long-distance" dependencies lead to interactions between neighbors that contradict those specified locally, commonly referred to as a "frustrated" model.

The experiments in this subsection repeat the procedure by which the results in Figure 5.8 were obtained, but using the four-node loopy model in Figure 5.9 with $w = 0.05$. In contrast to the four-node chain, we no longer expect successive iterations of belief propagation to converge to the true posterior marginals. Figure 5.10 shows the resulting average performance comparison, based again on 1000 samples from the measurement process Y. The network-constrained beliefs are seen to stabilize by the third stage, while those of belief propagation already begin to diverge, or oscillate. By making explicit use of memory, and through the offline optimization, our sequence of beliefs appears to be less susceptible to the so-called "double-counting" effect that confounds most other (online) message-passing algorithms when applied to loopy models.

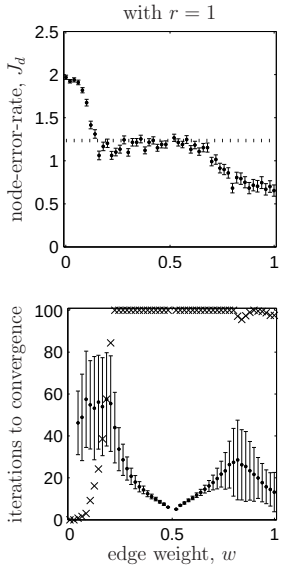
We close this chapter with some forward-looking speculation on the promise of our method as an alternative approximation paradigm in graphical models for which existing message-passing algorithms have difficulty. This comparison neglects the differences in communication overhead, in which our methods are superior by design. From the performance perspective, our approximation always provides an improvement over myopic performance, while loopy belief propagation (especially in the absence of convergence)



(a) "Loopy" Graphical Model



(b) Frustrated Regime of the Model



(c) Approximation via Belief Propagation

Figure 5.9. A (a) four-node graphical model with two cycles, (b) an illustration of its dependence on state correlation parameterized by w and (c) simulated decision performance (top figure) and convergence rate (bottom figure) of the loopy belief propagation algorithm across all values of w (and with measurement parameter $r = 1$, as usual). The algorithm performs reliably for edge weights above 0.5 (i.e., attractive models) and moderately below 0.5, but fails catastrophically in the "frustrated "regime (roughly $w < 0.25$), performing worse than the myopic strategy (with performance shown by the dotted horizontal line). The $\times$'s in the bottom figure indicate the percentage of Monte-Carlo runs in which usual sum-product converges before the 100th iteration, occurring infrequently for the same values of w in which performance is poor.

can fail catastrophically, performing even worse than the myopic approximation; on the other hand, when loopy belief propagation does converge, its performance is typically better than that of our network-constrained solutions. From the computational perspective, a clear disadvantage of our method is the offline overhead, an issue entirely absent in belief propagation. On the other hand, our online processing strategy is designed to terminate in only a few online iterations (by constraint), whereas the belief propagation algorithm in even small loopy models is seen to take an order of magnitude more iterations to converge (if it converges). In applications where convergence is difficult to guarantee over all probable measurements and online computation is orders of magnitude more expensive than offline computation, our methods become an attractive alternative. On the other hand, problems of practical interest will involve large graphical models, and whether our methods can scale in a manner comparable

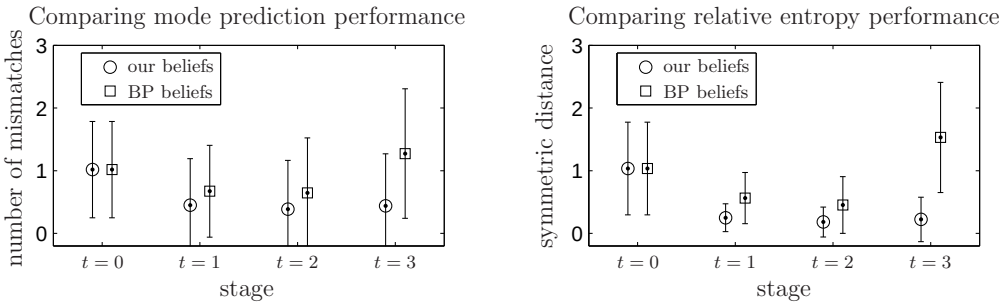


Figure 5.10. Empirical comparison between the sequence of "beliefs" (i.e., approximation of the true posterior marginals) produced by our network-constrained strategy and those produced by (unconstrained) belief propagation in a "frustrated" graphical model (i.e., model in Figure 5.9 with $w = 0.05$). Our network-constrained beliefs indicate improvement, under both error measures, over those produced by the (unconstrained) belief propagation algorithm.

to the scalability of belief propagation, while preserving the satisfactory performance demonstrated here for only the simplest models, remains to be seen.

Chapter 6

Conclusion

IN this thesis, motivated by the numerous engineering challenges associated with detection applications of so-called "collaborative self-organizing wireless sensor networks," we have formulated and analyzed an important class of network-constrained decision problems. The distinguishing assumption from their unconstrained counterparts is the presence of spatially-distributed decision objectives as well as explicit (and typically severe) constraints or costs on the available communication resources. Our introductory chapters drew connections between two traditionally separated active areas of research, namely approximate inference methods in graphical models and decentralized (team) Bayesian methods in multi-sensor detection. The complementary aspects of the associated models and algorithms led to our overarching hypothesis that the most promising distributed algorithmic solutions for sensor network applications lie at the intersection of these two areas. In the next section, we summarize the analysis and results of the preceding technical chapters in the context of how they support this hypothesis; the final section outlines the many questions that remain unanswered by this thesis in the context of recommendations for future research.

■ 6.1 Summary of Contributions

At the highest level, the contributions of this thesis can be stated in terms of applying well-understood ideas from the formalism of probabilistic graphical models into the formalism of decentralized detection models, and vice-versa. The manner in which this could be accomplished, however, depends upon a number of significant yet subtle assumptions about the global processing objectives and the network communication constraints. Arguably the most important of these from the engineering perspective is a crisp distinction between "online" measurement processing (i.e., the implementation of any collection of rules by which every node maps any particular measurement into its local decisions) and "offline" strategy optimization (i.e., the procedure by which all

rules are designed in order to mitigate the loss in global decision performance subject to the online network constraints). In particular, the value and feasibility of a self-organizing sensor network is not only measured by whether satisfactory online decision performance is achievable; it must also be the case that the network resources consumed for offline organization (and re-organization) represent only a modest fraction of the resources available over the total operational lifetime. Within the scope of this thesis, the decision objectives and network constraints are assumed to change slowly relative to the time intervals within which nodes are expected to receive measurements from the environment. In this case, the relatively high price of performing each offline organization can be amortized over a substantial number of highly-efficient online usages.

From an academic perspective, the contributions involve bridging the contrasting constraints and perspectives of the two disciplines of graphical models and decentralized decision-making. Recall that graphical models provide compact representations for the joint distribution of a large number of random variables, and the standard message-passing algorithms (e.g., belief propagation) exploit the graph structure to compute sufficient statistics efficiently for optimal decision-making. However, when each node in the graph is taken to be a spatially-distributed sensor, these message-passing algorithms effectively assume an ideal online communication model (e.g., a network medium featuring a reliable, high-rate link for every edge in the probability graph). On the other hand, decentralized detection models assume a non-ideal online communication model from the start (e.g., a low-rate or unreliable network medium), but the standard offline optimization algorithm requires that total computation/communication overhead scales exponentially with the number of nodes.

Altogether, standard approaches in graphical models require excessive online communication resources (but no need for offline organization), while standard approaches in decentralized detection lead to feasible online strategies (by constraint) but then require excessive offline network resources. The analysis and results in Chapter 3 show a simplest problem instance in which the best of both worlds is achieved: assuming (i) online measurement processing is constrained to a single forward sweep in a directed polytree network, (ii) the measurement/channel noise processes are spatially-independent and (iii) the global decision criterion decomposes additively across the nodes, the associated offline computation admits interpretation as an iterative forward-backward message-passing algorithm. Each forward sweep propagates likelihood messages, encoding what online communication along each link means from the transmitter's per-

spective, while each backward sweep propagates cost-to-go messages, encoding what online communication along each link means from the receiver's perspective. In each offline iteration, both types of incoming messages influence how each node updates its local rule parameters before it engages in the next iteration. The convergent offline iterations thus correspond to all nodes simultaneously arriving at a globally-consistent "fusion protocol" for how to both generate and interpret the communication symbols during subsequent online measurement processing.

The key steps by which we obtain these initial results can be traced to a collection of earlier works in the abundant decentralized (team) detection literature. As was discussed in Chapter 3, however, each of these earlier works considered only a special case of the model considered in Chapter 3, typically employing a proof technique not immediately applicable to our more general case. For example, our results hold for noisy channel models that include a dependence on the local hidden state (e.g., for detecting the presence or absence of a jamming signal) or the composite transmissions of all parent nodes (e.g., for modeling the effects of multipoint-to-point interference). Our results also shed new light on the extent to which the graphical structure underlying the spatially-distributed hidden state process may deviate from the communication network topology without sacrificing either algorithm correctness or efficiency. In particular, no matter the structure of the global prior probabilities $p(x)$, the offline message-passing algorithm assumes only that each node i is initialized with what we termed its neighborhood priors $p(x_i, x_{pa(i)})$, or the joint distribution of its own local state process and those of its parents $pa(i)$ on the *communication* graph.

Using essentially the same team-theoretic analysis techniques as in Chapter 3, subsequent technical chapters examine increasingly more elaborate online decision architectures. Our analysis goals remain the same: identify the minimum model assumptions under which we retain both application-layer correctness and network-layer efficiency in the developed algorithmic solutions (i.e., we can satisfy necessary team-optimality conditions via convergent offline message-passing algorithms). The first half of Chapter 4 considers the simplest architecture that introduces the prospect of bidirectional online communication, namely just one round of communication on an undirected network topology. Our analysis reveals a somewhat curious result: relative to what is known for the single-sweep directed architecture, the single-stage undirected architecture requires more assumptions to avoid worst-case intractability (i.e., when satisfying team-optimality conditions is NP-complete with even just two nodes), yet less assumptions to attain best-case tractability (i.e., when satisfying team-optimality conditions is

accomplished by an offline message-passing algorithm). The second half of Chapter 4 combines the two types of architectures, which we call hybrid networks, to introduce the prospect of long-distance communication among a subset of nodes. Under the usual model assumptions (i.e., spatially-independent noise, additive costs) and some mild restrictions on the interface between the two types of networks (e.g., a set of local directed networks, each with a single root, and an undirected network connecting these root nodes), we again show that team-optimality conditions can be satisfied with an offline message-passing algorithm.

The key step of our analyses in Chapter 4 is to "unravel" the bidirectional communication defined on the undirected or hybrid topology into an equivalent directed topology in which each node can appear as both a transmitter and a receiver. This simple idea has appeared in other research literature, referred to as a computation tree in the context of analyzing the "loopy" belief propagation algorithm [47, 63, 99] and a feedback architecture in the context of decentralized detection [3, 72]. Of course, as in Chapter 3, our problem differs from those treated by belief propagation in that the communication graph represents low-rate or unreliable links and need not bear any relation to the graph underlying the hidden state process. Our differences from the work on feedback architectures are more subtle. Firstly, the focus in this other work is on performing a global binary hypothesis test (rather than ours, which allows distributed objectives and decisions); secondly, it is assumed that each node processes only a new measurement in each stage, all nodes essentially "forgetting" all but a single bit of information about all previously-processed measurements. In contrast, our model assumes every node processes the same local measurement in successive decision stages, which in the undirected and hybrid architectures of Chapter 4 does not affect the applicability of our efficient message-passing algorithms.

The story changes dramatically for network-constrained decision architectures in which there are multiple stages of online communication. Drawing from the canonical message schedules employed in belief propagation, Chapter 5 formulates multi-stage architectures for both directed and undirected network topologies, the former consisting of repeated forward-backward sweeps and the latter consisting of repeated parallel exchanges. Our team-theoretic analysis exposes a number of new structural properties that an optimal multi-stage processing strategy should satisfy, including how the use of memory at each node affords an increasingly accurate approximation to its posterior marginals (i.e., the sufficient statistic for making its local state-related decision that sum-product belief propagation aims to compute). Unfortunately, even under best-case

model assumptions, the required memory (and, in turn, the offline solution complexity) grows exponentially with the number of online communication stages. Nonetheless, an approximation that leverages repeated application of the efficient single-stage solutions demonstrates appealing empirical results in comparison to unconstrained belief propagation algorithms on several small-scale models.

We may sum up the academic contributions of this thesis as follows. From the perspective of probabilistic graphical models, we developed new online message-passing algorithms in which global decision performance degrades gracefully as network constraints become arbitrarily severe. These constraints include a fixed small number of iterations, the presence of low-rate or unreliable links, or a communication graph that differs from the underlying probability graph. From the perspective of decentralized detection models, we developed new offline message-passing algorithms that remain tractable for a larger class of detection objectives and network constraints than previously considered. This class of problems includes explicit communication-related costs as well as the usual detection-related costs but with spatially-distributed hidden state processes and perhaps multiple gateway (i.e., decision-making) nodes; it also extends to unreliable networks defined on either directed and undirected topologies as well as certain combinations of the two.

■ 6.2 Recommendations for Future Research

There are a variety of open research problems arising from this thesis. Some involve strengthening the established convergence/efficiency guarantees we've obtained, others involve relaxing one or more of the modeling assumptions we've made, and still others involve designing entirely new (and ideally distributed) algorithms for obtaining quantities that our offline message-passing solutions consider to be given. We categorize the recommendations for future research into whether or not there is only one stage of online communication, reflecting the fundamental divide in complexity for multi-stage architectures exposed during the course of this thesis.

■ 6.2.1 Single-Stage Communication Architectures

Recall that the theoretical convergence guarantee for the offline message-passing algorithm in Chapter 3 is only with respect to the penalty sequence $\{J(\gamma^k)\}$. However, empirically, we have yet to observe the associated parameter sequence $\{\theta^k\}$ itself fail to converge, but the possibility is known to exist for coordinate-descent algorithms, in general. This begs the question as to whether the assumptions by which the offline

message-passing is derived also allow for stronger convergence statements than those inherited from the more general case considered in Corollary 3.1. One line of attack could be to establish that the offline message-passing equations are contractions under some distance metric between successive iterates [6, 71]. Similar questions arise as to whether the offline message-passing algorithm is more amenable to bounds on the achievable decentralized performance: while we have used the (zero communication) myopic upper bound and the (infeasible) centralized lower bound to gauge the success of our solutions, we have no results on the performance relative to that of the best *feasible* strategy (i.e., not one constrained only to be person-by-person optimal).

Another important category of questions concerns the robustness of the offline message-passing algorithm when not every assumption under which it is convergent can be satisfied. One such question is the degree to which errors (e.g., due to high-rate quantization) in the offline messages can be tolerated. Analogous questions have been studied for the belief propagation message-passing algorithms [47, 90], but the key difference in our setup is that there are two different types of messages and, moreover, the rule parameters also change with successive iterations. A similar line of questioning could bound the adverse effects of mismatches between the local models assumed at any particular node from the true ones. This is especially pertinent as concerns the neighborhood priors $p(x_i, x_{pa(i)})$ in directed networks or $p(x_i, x_{ne(i)})$ in undirected networks, which may themselves be difficult to compute exactly when the communication graph is radically different from the probability graph. It is also of keen interest with respect to the rule-dependent statistics $p(u_i, \hat{x}_i | x_i, u_{pa(i)}; \gamma_i)$ local to each node i, as the associated marginalization over Y_i can be difficult to carry out exactly in certain measurement models of practical interest.

The experiments in Chapter 3 only scratched the surface of the many robustness questions with respect to whether the required model assumptions are satisfied. In particular, we compared our message-passing solution to the true team-optimal solution in a simplest non-tree-structured detection network. While performance of the tree-based approximation was notably inferior, it still performed well relative to the benchmark myopic/centralized performances. Other interesting questions along these lines is how much is lost when not all noise processes are spatially independent, or when the cost function does not decompose additively across the nodes. Part of addressing these questions in larger examples could require a network-constrained analog to the junction-tree algorithm [49, 60], where multiple nodes must be merged into super-nodes before one can tractably compute the true team-optimal solution. Understanding such robustness

properties is also a first step towards addressing the even more difficult problem of when a detection network should reorganize i.e., when have the network topology, the decision objective or the local models changed enough to merit re-optimization as opposed to just accepting the potentially degraded online performance using the rules obtained from the preceding optimization.

Finally, in all of our analysis and almost all of our examples, we assumed that the network topology and the gateway nodes were given. The one exception was when we were randomly generating 100-node detection networks for our large-scale experiments in Chapter 3 and Chapter 4. There we employed a simple heuristic based on the given probabilistic model and neglecting the need for a distributed algorithm to do so, solving for a max-weight spanning tree using as weights the pairwise correlations between the hidden state variables. Optimizing the selection of the topology and desirable gateway nodes is the subject of ongoing research [91]. Another interesting extension to our model would be to equip certain nodes with the option to request additional information from its neighbors, perhaps with some additional cost, as has been studied so far (to our knowledge) only in a simplest two-node tandem network [74].

■ 6.2.2 Multi-Stage Communication Architectures

In comparison to single-stage architectures, our understanding of multi-stage architectures is far more limited and, in turn, our suggestions for future research are less specific. One particularly obvious suggestion is a proof or disproof of Conjecture 5.1, although our approximation that takes it to be true has shown in preliminary experiments that it may be a sound assumption regardless. Of course, more comprehensive experimentation is required to say for certain. For more realistic problems, it may also turn out that neglecting the true dependence of each node's stage-t side information on all nodes' preceding communication rules is too simplistic. New methods for exploiting the causal processing assumptions and perhaps other available structure in the prior probabilities $p(x)$ may lead to performance gains that are worth the additional offline computation overhead.

There are many other facets to the exposed problem complexity that have not been tackled satisfactorily by our approximate solution method. The main one is the exponential complexity in the parameterization of the online processing rules. Our experiments have so far considered only small-scale problems in which the this complexity is not yet the main barrier. However, based on intuition associated with inference in graphical models, we'd like to push towards a number of stages on the order of the di-

ameter of the probability graph. In these cases, methods for systematically reducing the memory requirements, perhaps adaptively as a function of all observed data, become crucial. The most promising methods for such approximation may show themselves in the limit of infinite-horizon analyses, similar to how steady-state approximations to finite-horizon control problems often provide satisfactory approximate solutions.

Appendix A

Directed Network Constraints: Proofs

■ A.1 Person-by-Person Optimality

Proposition 3.1 is proven as follows. The rule γ_i^* minimizes J in (3.1) over all Γ_i, holding all other rules fixed at $\gamma_{\setminus i}^*$, if and only if the process $(U_i, \hat{X}_i) = \gamma_i(Y_i, Z_i)$ minimizes

$$\mathbf{E}\left[c\left(U_{\setminus i}, u_i, \hat{X}_{\setminus i}, \hat{x}_i, X\right) \middle| Y_i, Z_i; \gamma_{\setminus i}^* \right], \tag{A.1}$$

over all possible realizations $(u_i, \hat{x}_i) \in \mathcal{U}_i \times \mathcal{X}_i$, with probability one. Fix a realization $(u_i, \hat{x}_i)$ and consider the distribution $p(u_{\setminus i}, \hat{x}_{\setminus i}, x | y_i, z_i; \gamma_{\setminus i}^*, u_i, \hat{x}_i)$ underlying (A.1), or equivalently

$$p(u_{\setminus i}, \hat{x}_{\setminus i} | x, y_i, z_i; \gamma_{\setminus i}^*, u_i, \hat{x}_i) p(x | y_i, z_i; \gamma_{\setminus i}^*, u_i, \hat{x}_i).$$

By virtue of Lemma 3.1, the first term simplifies to

$$p(u_{\setminus i}, \hat{x}_{\setminus i} | x, y_i, z_i; \gamma_{\setminus i}^*, u_i) = \frac{p(u_{\setminus i}, z_i, \hat{x}_{\setminus i} | x; \gamma_{\setminus i}^*, u_i)}{p(z_i | x; \gamma_{\setminus i}^*)},$$

and, applying Bayes' rule, the second term simplifies to

$$p(x | y_i, z_i; \gamma_{\setminus i}^*) = \frac{p(x) p(y_i | x) p(z_i | x; \gamma_{\setminus i}^*)}{p(y_i, z_i; \gamma_{\setminus i}^*)}$$

for every $z_i \in \mathcal{Z}_i$ such that $p(y_i, z_i; \gamma_{\setminus i}^*) > 0$. Taking the product of the two fractions, the positive-valued denominator neither depends on x nor on $(u_i, \hat{x}_i)$ and, as such, has no bearing on the minimization of (A.1). Altogether, it suffices to require that $\gamma_i(Y_i, z_i)$ minimize

$$\sum_{x \in \mathcal{X}} \theta_i^*(u_i, \hat{x}_i, x; z_i) p(Y_i | x)$$

with probability one, where for each fixed value of $(u_i, \hat{x}_i)$,

$$\theta_i^*(u_i, \hat{x}_i, x; z_i) = \sum_{u_{\setminus i}} \sum_{\hat{x}_{\setminus i}} c(u, \hat{x}, x) p(u_{\setminus i}, z_i, \hat{x}_{\setminus i}, x; \gamma_{\setminus i}^*, u_i) \tag{A.2}$$

and, again by virtue of Lemma 3.1,

$$p(u_{\setminus i}, z_i, \hat{x}_{\setminus i}, x; \gamma_{\setminus i}^*, u_i) = p(x) p(z_i | x, u_{pa(i)}) \prod_{j \neq i} p(u_j, \hat{x}_j | x, u_{pa(j)}; \gamma_j^*).$$

■ A.2 Offline Efficiency

Proposition 3.2 is proven as follows. With Assumption 3.2 in effect, we may begin with the person-by-person optimality conditions expressed in Corollary 3.2. With Assumption 3.3 also in effect, we may substitute (3.15) into (3.1), obtaining for any fixed strategy $\gamma \in \Gamma$ an additive global penalty function,

$$J(\gamma) \quad = \quad \sum_{i=1}^{n} G_i(\gamma)$$

with

$$G_i(\gamma) \quad = \quad \sum_{x_i} p(x_i) \sum_{u_i} \sum_{\hat{x}_i} c(u_i, \hat{x}_i, x_i) \sum_{z_i} p(z_i | x_i; \gamma) p(u_i, \hat{x}_i | x_i, z_i; \gamma_i)$$

for each i, where we have employed the identities

$$p(u_i, \hat{x}_i, x_i; \gamma) = p(x_i) \sum_{z_i} p(z_i, u_i, \hat{x}_i | x_i; \gamma) = p(x_i) \sum_{z_i} p(z_i | x_i; \gamma) p(u_i, \hat{x}_i | x_i, z_i; \gamma_i).$$

Lemma A.1. *Let Assumption 3.2 and Assumption 3.3 hold. Then Corollary 3.2 applies with (3.13) specialized to*

$$\phi_i^*(u_i, \hat{x}_i, x_i; z_i) \propto p(x_i) P_i^*(z_i | x_i) \left[c(u_i, \hat{x}_i, x_i) + C_i^*(u_i, x_i; z_i) \right]$$

with likelihood function

$$P_i^*(z_i | x_i) = p(z_i | x_i; \gamma_{\setminus i}^*)$$

and cost-to-go function

$$C_i^*(u_i, x_i; z_i) = \sum_{m \in de(i)} \sum_{x_m} \sum_{u_m} \sum_{\hat{x}_m} p(x_m, u_m, \hat{x}_m | z_i, u_i, x_i; \gamma_{\setminus i}^*) c(u_m, \hat{x}_m, x_m),$$

where $de(i)$ denotes the descendants *of node i (i.e., the children $ch(i)$, each such child's children, and so on) in the directed network $\mathcal{F}$.*

Proof. Substitute (3.15) into (A.2) and rearrange summations to obtain

$$\theta_i^*(u_i, \hat{x}_i, x; z_i) = p(x, z_i; \gamma_{\backslash i}^*) \left[c(u_i, \hat{x}_i, x_i) + \right.$$
$$\left. \sum_{m \neq i} \sum_{u_m} \sum_{\hat{x}_m} p(u_m, \hat{x}_m | x, z_i, u_i; \gamma_{\backslash i}^*) c(u_m, \hat{x}_m, x_m) \right].$$

Conditioned on $Z_i = z_i$, the penalty term for each m other than the local node i or any one of its descendants $de(i)$ will not depend upon the candidate decision $(u_i, \hat{x}_i)$, so each such term has no bearing on the minimization in (3.8). That is, in Proposition 3.1 it now suffices to satisfy

$$\theta_i^*(u_i, \hat{x}_i, x; z_i) \propto p(x, z_i; \gamma_{\backslash i}^*) \left[c(u_i, \hat{x}_i, x_i) + \right.$$
$$\left. \sum_{m \in de(i)} \sum_{u_m} \sum_{\hat{x}_m} p(u_m, \hat{x}_m | x, z_i, u_i; \gamma_{\backslash i}^*) c(u_m, \hat{x}_m, x_m) \right]$$

and, in turn, in Corollary 3.2 it now suffices to satisfy

$$\begin{aligned} \phi_i^*(u_i, \hat{x}_i, x_i; z_i) &\propto \sum_{x_{\backslash i}} p(x, z_i; \gamma_{\backslash i}^*) \left[c(u_i, \hat{x}_i, x_i) + \right. \\ &\qquad \left. \sum_{m \in de(i)} \sum_{u_m} \sum_{\hat{x}_m} p(u_m, \hat{x}_m | x, z_i, u_i; \gamma_{\backslash i}^*) c(u_m, \hat{x}_m, x_m) \right] \\ &= p(x_i, z_i | \gamma_{\backslash i}^*) \left[c(u_i, \hat{x}_i, x_i) + \right. \\ &\left. \sum_{x_{\backslash i}} p(x_{\backslash i} | x_i, z_i; \gamma_{\backslash i}^*) \sum_{m \in de(i)} \sum_{u_m} \sum_{\hat{x}_m} p(u_m, \hat{x}_m | x, z_i, u_i; \gamma_{\backslash i}^*) c(u_m, \hat{x}_m, x_m) \right] \\ &= p(x_i) P_i^*(z_i | x_i) \left[c(u_i, \hat{x}_i, x_i) + C_i^*(u_i, x_i; z_i) \right], \end{aligned}$$

the last line employing the identity

$$\sum_{x_{\backslash i}} p(x_{\backslash i} | x_i, z_i; \gamma_{\backslash i}^*) p(u_m, \hat{x}_m | x, z_i, u_i; \gamma_{\backslash i}^*) = p(u_m, \hat{x}_m | x_i, z_i, u_i; \gamma_{\backslash i}^*)$$

for every descendant $m \in de(i)$. □

Lemma A.2. *Let Assumption 3.2 and Assumption 3.4 hold. Then, under any fixed strategy $\gamma \in \Gamma$, the* local *likelihood function for received information Z_i at each node i (with at least one ancestor) satisfies*

$$p(z_i|x_i;\gamma) \propto \sum_{u_{pa(i)}} p(z_i|x_i,u_{pa(i)}) \sum_{x_{pa(i)}} p(x_{pa(i)}|x_i) \prod_{j\in pa(i)} p(u_j|x_j;\gamma)$$

with

$$p(u_j|x_j;\gamma) = \sum_{z_j} p(z_j|x_j;\gamma) \sum_{\hat{x}_j} p(u_j,\hat{x}_j|x_j,z_j;\gamma_j)$$

for every parent $j \in pa(i)$.

Proof. Let $an(i)$ denote the ancestors of node i (i.e., the parents $pa(i)$ of node i, each such parent's parents, and so on). Starting from Corollary 3.2, for every node i without ancestors (and hence without information Z_i), we have $p(z_i|x;\gamma) = 1$ and $p(u_i,\hat{x}_i|x,z_i;\gamma) = p(u_i,\hat{x}_i|x_i;\gamma_i)$. For every node i with ancestors, the forward partial order of network topology $\mathcal{F}$ implies the recursive definition

$$\begin{aligned}
p(z_i|x;\gamma) &= \sum_{z_{pa(i)}} \sum_{u_{pa(i)}} \sum_{\hat{x}_{pa(i)}} p(z_{pa(i)},u_{pa(i)},\hat{x}_{pa(i)},z_i|x;\gamma) \\
&= \sum_{u_{pa(i)}} p(z_i|x_i,u_{pa(i)}) \sum_{z_{pa(i)}} \sum_{\hat{x}_{pa(i)}} p(z_{pa(i)},u_{pa(i)},\hat{x}_{pa(i)}|x;\gamma) \\
&= \sum_{u_{pa(i)}} p(z_i|x_i,u_{pa(i)}) \sum_{z_{pa(i)}} p(z_{pa(i)}|x;\gamma) \sum_{\hat{x}_{pa(i)}} p(u_{pa(i)},\hat{x}_{pa(i)}|x,z_{pa(i)};\gamma) \\
&= \sum_{u_{pa(i)}} p(z_i|x_i,u_{pa(i)}) \sum_{z_{pa(i)}} p(z_{pa(i)}|x_{an(i)};\gamma_{an(i)-pa(i)}) \times \\
&\qquad \prod_{j\in pa(i)} \sum_{\hat{x}_j} p(u_j,\hat{x}_j|x_j,z_j;\gamma_j) \\
&\equiv p(z_i|x_{an(i)},x_i;\gamma_{an(i)}). \qquad \text{(A.3)}
\end{aligned}$$

We see that the global likelihood function for information Z_i received by each node i from its parents $pa(i)$ (if any) depends at most on the rules $\gamma_{an(i)}$ local to all ancestors and the states $(X_{an(i)}, X_i)$ local to itself and its ancestors. In turn, the global likelihood function for information U_i transmitted by each node i to its children $ch(i)$ (if any) is

$$\begin{aligned}
p(u_i|x;\gamma) &= \sum_{z_i} p(z_i|x;\gamma) \sum_{\hat{x}_i} p(u_i,\hat{x}_i|x_i,z_i;\gamma_i) \\
&\equiv p(u_i|x_{an(i)},x_i;\gamma_{an(i)},\gamma_i). \qquad \text{(A.4)}
\end{aligned}$$

Now, Assumption 3.4 ensures that no two nodes have a common ancestor, or equivalently that the collection of index sets $\{an(j); j \in pa(i)\}$ partition the index set $an(i) - pa(i)$. Because individual measurements are assumed to be mutually independent (conditioned on X), information derived from mutually-exclusive subsets of measurements will be similarly independent i.e.,

$$p(z_{pa(i)}|x;\gamma) = \prod_{j \in pa(i)} p(z_j|x;\gamma). \tag{A.5}$$

Combining (A.3)–(A.5) yields

$$\begin{aligned} p(z_i|x;\gamma) &= \sum_{u_{pa(i)}} p(z_i|x_i, u_{pa(i)}) \times \\ &\qquad \prod_{j \in pa(i)} \left(\sum_{z_j} p(z_j|x_{an(j)}, x_j; \gamma_{an(j)}) \sum_{\hat{x}_j} p(u_j, \hat{x}_j|x_j, z_j; \gamma_j) \right) \\ &= \sum_{u_{pa(i)}} p(z_i|x_i, u_{pa(i)}) \prod_{j \in pa(i)} p(u_j|x;\gamma), \end{aligned}$$

so that

$$\begin{aligned} p(z_i|x_i;\gamma) &= \sum_{x_{\setminus i}} p(x_{\setminus i}|x_i) p(z_i|x;\gamma) \\ &= \sum_{u_{pa(i)}} p(z_i|x_i, u_{pa(i)}) \sum_{x_{an(i)}} p(x_{an(i)}|x_i) \prod_{j \in pa(i)} p(u_j|x;\gamma) \\ &= \sum_{u_{pa(i)}} p(z_i|x_i, u_{pa(i)}) \sum_{x_{pa(i)}} p(x_{pa(i)}|x_i) \times \\ &\qquad \sum_{x_{an(i)\setminus pa(i)}} p(x_{an(i)-pa(i)}|x_{pa(i)}, x_i) \prod_{j \in pa(i)} p(u_j|x;\gamma). \end{aligned} \tag{A.6}$$

It remains to show that the inner sum in (A.6) is proportional to

$$\prod_{j \in pa(i)} \sum_{x_{an(j)}} p(x_{an(j)}|x_j) p(u_j|x;\gamma) = \prod_{j \in pa(i)} p(u_j|x_j;\gamma), \tag{A.7}$$

where each jth factor is seen to be equal to $p(u_j|x_j;\gamma)$ by virtue of (A.4). First recognize that, for any particular $m \in pa(i)$, we may write

$$p(x_{an(i)-pa(i)}|x_{pa(i)}, x_i) = p(x_{an(i)-pa(i)-an(m)}|x_{pa(i)}, x_i) p(x_{an(m)}|x_{an(i)-an(m)}, x_i),$$

in which case the inner sum in (A.6) is equivalent to

$$p(u_m|x_{an(i)-an(m)}, x_i; \gamma) \sum_{x_{an(i)-pa(i)-an(m)}} p(x_{an(i)-pa(i)-an(m)}|x_{pa(i)}, x_i) \times \prod_{j \in pa(i) \setminus m} p(u_j|x; \gamma)$$

with

$$\begin{aligned} p(u_m|x_{\alpha(i)-\alpha(m)}, x_i; \gamma) &= \sum_{x_{\alpha(m)}} p(x_{\alpha(m)}|x_{\alpha(i)-\alpha(m)}, x_i) p(u_m|x; \gamma) \\ &= \sum_{x_{\alpha(m)}} \left(\frac{p(x_{\alpha(i)}, x_i|x_m)}{p(x_{\alpha(i)-\alpha(m)}, x_i|x_m)} \right) p(u_m|x; \gamma) \\ &= \frac{\sum_{x_{\alpha(m)}} p(x_{\alpha(i)}, x_i|x_m) p(u_m|x; \gamma)}{p(x_{\alpha(i)-\alpha(m)}, x_i|x_m)} \\ &\propto \sum_{x_{\alpha(m)}} p(x_{\alpha(m)}|x_m) p(u_m|x; \gamma). \end{aligned}$$

For any other parent $\ell \in \pi(i) - m$, if we let $an(m, \ell)$ denote the union $an(m) \cup an(\ell)$, we may similarly write

$$p(x_{an(i)-pa(i)-an(m)}|x_{pa(i)}, x_i) = p(x_{an(i)-pa(i)-an(m,\ell)}|x_{pa(i)}, x_i) p(x_{an(\ell)}|x_{an(i)-an(m,\ell)}, x_i)$$

and conclude that the inner sum in (A.6) is equivalent to

$$p(u_m|x_{an(i)-an(m)}, x_i; \gamma) p(u_\ell|x_{an(i)-an(\ell)}, x_i; \gamma) \times \sum_{x_{an(i)-pa(i)-an(m,\ell)}} p(x_{an(i)-pa(i)-an(m,\ell)}|x_{pa(i)}, x_i) \prod_{j \in pa(i) \setminus \{m,\ell\}} p(u_j|x; \gamma)$$

with

$$p(u_\ell|x_{an(i)-an(\ell)}, x_i; \gamma) \propto \sum_{x_{an(\ell)}} p(x_{an(\ell)}|x_\ell) p(u_\ell|x; \gamma).$$

Continuing this procedure on a parent-by-parent basis brings us to (A.7). □

Taken together, Lemma A.1 and Lemma A.2 lead directly to the forward likelihood recursions in Proposition 3.2. The backward cost-to-go recursions also result from Lemma A.1 and Lemma A.2, taken alongside a couple of additional arguments. Firstly, by virtue of Assumption 3.4, the one path from any ancestor of node i to any descendant

of node i includes node i. So, when conditioning on received information $Z_i = z_i$ and holding local decision $(u_i, \hat{x}_i)$ fixed, the information already received and transmitted by all ancestors is independent (conditioned on X) of the information to be received and transmitted by all descendents; mathematically, for each descendant $m \in de(i)$ in Lemma A.1, we have

$$\begin{aligned} p(u_m, \hat{x}_m | x, z_i, u_i; \gamma^*_{\setminus i}) &= p(u_m, \hat{x}_m | x, u_i; \gamma^*_{an(m)\setminus i - an(i)}, \gamma^*_m) \\ \Rightarrow \quad C^*_i(u_i, x_i; z_i) &= C^*_i(u_i, x_i) \end{aligned}$$

and, in turn, the pbp-optimal parameter values ϕ^*_i specialize to the form in (3.16). Secondly, Assumption 3.4 also guarantees no two children have a common descendant, implying that downstream costs decompose additively across child nodes i.e., for each i,

$$\sum_{j \in de(i)} G_j(\gamma) = \sum_{j \in ch(i)} \left[G_j(\gamma) + \sum_{m \in de(j)} G_m(\gamma) \right].$$

Appendix B

Undirected Network Constraints: Proofs

■ B.1 Person-by-Person Optimality

Proposition 4.1 is proven as follows. Firstly, with Assumption 4.1 in effect, the analogous steps taken in the proof to Lemma 3.1 conclude that, for every strategy $\gamma \in \Gamma$, the distribution in (4.1) specializes to

$$p(u, \hat{x}, x; \gamma) = p(x) \prod_{i=1}^{n} p(u_i, \hat{x}_i | x, u_{ne(i)}; \gamma_i), \tag{B.1}$$

where for every i,

$$p(u_i, \hat{x}_i | x, u_{ne(i)}; \gamma_i) = \sum_{z_i \in \mathcal{Z}_i} p(z_i | x, u_{ne(i)}) \int_{y_i \in \mathcal{Y}_i} p(y_i | x) p(u_i, \hat{x}_i | y_i, z_i; \gamma_i) \, dy_i.$$

Each item in Proposition 4.1 is then proven via analogous steps to those taken in the proof to Proposition 3.1.

- The stage-two rule δ_i^* minimizes J in (3.1) over all Δ_i, holding the local stage-one rule and the rules local to all other nodes fixed, if and only if the process $\hat{X}_i = \delta_i(Y_i, U_i, Z_i)$ minimizes

$$E\left[c(U, \hat{X}_{\setminus i}, \hat{x}_i, X) | Y_i, U_i, Z_i; \mu_i^*, \gamma_{\setminus i}^*\right], \tag{B.2}$$

  over all possible realizations $\hat{x}_i \in \mathcal{X}_i$, with probability one. Fix a realization $\hat{x}_i$ and consider the distribution $p(u, \hat{x}_{\setminus i}, x | y_i, u_i, z_i; \mu_i^*, \gamma_{\setminus i}^*, \hat{x}_i)$ underlying (B.2), or equivalently

$$p(u, \hat{x}_{\setminus i} | x, y_i, u_i, z_i; \mu_i^*, \gamma_{\setminus i}^*, \hat{x}_i) p(x | y_i, u_i, z_i; \mu_i^*, \gamma_{\setminus i}^*, \hat{x}_i).$$

By virtue of (B.1), for every $(u_i, z_i) \in \mathcal{U}_i \times \mathcal{Z}_i$ such that $p(y_i, u_i, z_i; \mu_i^*, \gamma_{\setminus i}^*) > 0$, the first term simplifies to

$$\begin{aligned} p(u, \hat{x}_{\setminus i}|x, y_i, u_i, z_i; \mu_i^*, \gamma_{\setminus i}^*) &= p(u_i|y_i, u_i; \mu_i^*) p(u_{\setminus i}, \hat{x}_{\setminus i}|x, u_i, z_i; \gamma_{\setminus i}^*) \\ &= \frac{p(u_{\setminus i}, z_i, \hat{x}_{\setminus i}|x, u_i; \gamma_{\setminus i}^*)}{p(z_i|x; \gamma_{\setminus i}^*)}, \end{aligned}$$

and the second term simplifies to

$$p(x|y_i, u_i, z_i; \mu_i^*, \gamma_{\setminus i}^*) = \frac{p(x) p(y_i, u_i|x; \mu_i^*) p(z_i|x; \gamma_{\setminus i}^*)}{p(y_i, u_i, z_i; \mu_i^*, \gamma_{\setminus i}^*)}.$$

Taking the product of the two fractions, the positive-valued denominator neither depends on x nor on $\hat{x}_i$ and, as such, has no bearing on the minimization of (A.1); moreover, with the (deterministic) stage-one rule fixed at μ_i^*, it follows that

$$p(y_i, u_i|x; \mu_i^*) \propto \begin{cases} p(y_i|x) & , \text{ if } u_i = \mu_i^*(y_i) \\ 0 & , \text{ otherwise} \end{cases}.$$

Altogether, it suffices to require that $\hat{X}_i = \delta_i(Y_i, u_i, z_i)$ minimize

$$\sum_{x \in \mathcal{X}} b_i^*(\hat{X}_i, x; u_i, z_i) p(Y_i|x)$$

with probability one, where for each candidate decision $\hat{x}_i$,

$$b_i^*(\hat{x}_i, x; u_i, z_i) = \sum_{u_{\setminus i}} \sum_{\hat{x}_{\setminus i}} c(u, \hat{x}, x) p(x) p(u_{\setminus i}, z_i, \hat{x}_{\setminus i}|x, u_i; \gamma_{\setminus i}^*) \tag{B.3}$$

and, again by virtue of Lemma 3.1,

$$p(u_{\setminus i}, z_i, \hat{x}_{\setminus i}|x, u_i; \gamma_{\setminus i}^*) = p(z_i|x, u_{ne(i)}) \prod_{j \neq i} p(u_j, \hat{x}_j|x, u_{ne(j)}; \gamma_j^*).$$

- The stage-one rule μ_i^* minimizes J in (3.1) over all $\mathcal{M}_i$, holding the local stage-two rule and the rules local to all other nodes fixed, if and only if the process $U_i = \mu_i(Y_i)$ minimizes

$$E\left[c(U_{\setminus i}, u_i, \hat{X}, X)|Y_i; \delta_i^*, \gamma_{\setminus i}^*\right], \tag{B.4}$$

over all possible realizations $u_i \in \mathcal{U}_i$, with probability one. Fix a realization u_i and consider the distribution $p(u_{\setminus i}, \hat{x}, x|y_i; \delta_i^*, \gamma_{\setminus i}^*, u_i)$ underlying (B.4), or equivalently

$$p(u_{\setminus i}, \hat{x}|x, y_i; \delta_i^*, \gamma_{\setminus i}^*, u_i) p(x|y_i; \delta_i^*, \gamma_{\setminus i}^*, u_i).$$

By virtue of (B.1), the first term simplifies to

$$p(\hat{x}_i|y_i, u_i, u_{ne(i)}; \delta_i^*) \prod_{j \neq i} p(u_j, \hat{x}_j|x, u_{ne(j)}; \gamma_j^*)$$

and, because knowledge of decision u_i implies nothing about the measurement y_i when we consider rule μ_i subject to design, the second term is equivalent to $p(x|y_i)$. We also have the identity

$$p(\hat{x}_i|y_i, u_i, u_{ne(i)}; \delta_i^*) = \sum_{z_i \in \mathcal{Z}_i} p(z_i|x, u_{ne(i)}) p(\hat{x}_i|y_i, u_i, z_i; \delta_i^*)$$

and, By Bayes' rule, $p(x|y_i)$ is proportional to $p(x)p(y_i|x)$ for every y_i. Altogether, it suffices to require that $U_i = \mu_i(Y_i)$ minimize

$$\sum_{x \in \mathcal{X}} a_i^*(U_i, x; Y_i) p(Y_i|x)$$

with probability one, where for each observed value of y_i and candidate decision u_i,

$$a_i^*(u_i, x; y_i) = \sum_{u_{\setminus i}} \sum_{\hat{x}} c(u, \hat{x}, x) p(x) p(u_{\setminus i}, \hat{x}|x, y_i; \delta_i^*, \gamma_{\setminus i}^*, u_i) \tag{B.5}$$

with

$$p(u_{\setminus i}, \hat{x}|x, y_i; \delta_i^*, \gamma_{\setminus i}^*, u_i) = \left(\sum_{z_i} p(z_i|x, u_{ne(i)}) p(\hat{x}_i|y_i, u_i, z_i; \delta_i^*) \right) \prod_{j \neq i} p(u_j, \hat{x}_j|x, u_{ne(j)}; \gamma_j^*).$$

■ B.2 Tractable Person-by-Person Optimality

Proposition 4.2 is proven as follows. We start with the stage-two decision rule in Proposition 4.1, where substitution of (4.6) into (B.3) gives

$$\begin{aligned} b_i^*(\hat{x}_i, x; u_i, z_i) &= p(x) \sum_{u_{\setminus i}} \sum_{\hat{x}_{\setminus i}} \left[\sum_{m=1}^{n} c(\hat{x}_m, x) + \lambda c(u_m, x) \right] p(u_{\setminus i}, z_i, \hat{x}_{\setminus i}|x, u_i; \gamma_{\setminus i}^*) \\ &= p(x) \left[c(\hat{x}_i, x) + \lambda c(u_i, x) \right] p(z_i|x, u_i; \gamma_{\setminus i}^*) + \\ &\qquad p(x) \sum_{m \neq i} \left[c(\hat{x}_m, x) + \lambda c(u_m, x) \right] p(u_m, z_i, \hat{x}_m|x, u_i; \gamma_{\setminus i}^*). \end{aligned}$$

Only the first part of the ith term depends upon candidate decision $\hat{x}_i$, and all other terms thus have no bearing on the minimization in (4.4). That is, for the stage-two rule in Proposition 4.1, it suffices to satisfy

$$b_i^*(\hat{x}_i, x; u_i, z_i) \propto p(x)c(\hat{x}_i, x)p(z_i|x, u_i; \gamma_{\setminus i}^*)$$

and, by virtue of (B.1),

$$\begin{aligned} p(z_i|x, u_i; \gamma_{\setminus i}^*) &= p(z_i|x; \gamma_{\setminus i}^*) \\ &= \sum_{u_{ne(i)}} p(u_{ne(i)}, z_i|x; \gamma_{\setminus i}^*) \\ &= \sum_{u_{ne(i)}} p(z_i|x, u_{ne(i)})p(u_{ne(i)}|x; \gamma_{\setminus i}^*) \\ &= \sum_{u_{ne(i)}} p(z_i|x, u_{ne(i)}) \prod_{j \in ne(i)} p(u_j|x; \mu_j^*), \end{aligned}$$

where in the last step we have employed the identity

$$\begin{aligned} p(u_j|x; \gamma_{\setminus i}^*) &= \sum_{z_j} p(z_j|x; \gamma_{\setminus i}^*) \sum_{\hat{x}_j} p(u_j, \hat{x}_j|x, z_j; \gamma_j^*) \\ &= \sum_{z_j} p(z_j|x; \gamma_{\setminus i}^*) \sum_{\hat{x}_j} \int_{y_j \in \mathcal{Y}_j} p(u_j|y_j; \mu_j^*)p(\hat{x}_j|y_j, u_j, z_j; \delta_j^*)p(y_j|x)dy_j \\ &= \int_{y_j \in \mathcal{Y}_j} p(u_j|y_j; \mu_j^*)p(y_j|x)dy_j \quad = \quad p(u_j|x; \mu_j^*). \end{aligned}$$

Observe that parameters b_i^*, and hence the stage-two rule δ_i^*, no longer depend upon the local stage-one decision u_i. In other words, the optimal local stage-two rule (assuming all other rules fixed) lies in the subset of Δ_i consisting of all functions of the form $\delta_i : \mathcal{Y}_i \times \mathcal{Z}_i \to \mathcal{X}_i$ and, in turn, we may assume without loss of generality that $p(\hat{x}_i|y_i, u_i, z_i; \delta_i^*) = p(\hat{x}_i|y_i, z_i; \delta_i^*)$. Applying this same reduction to the local stage-two rule δ_j^* of every other node, we have the identity

$$\begin{aligned} p(\hat{x}_j|x, z_j; \gamma_j^*) &= \sum_{u_j} p(u_j, \hat{x}_j|x, z_j; \gamma_j^*) \\ &= \sum_{u_j} \int_{y_j \in \mathcal{Y}_j} p(u_j|y_j; \mu_j^*)p(\hat{x}_j|y_j, z_j; \delta_j^*)p(y_j|x)\, dy_j \\ &= \int_{y_j \in \mathcal{Y}_j} p(\hat{x}_j|y_j, z_j; \delta_j^*)p(y_j|x)\, dy_j \quad = \quad p(\hat{x}_j|x, z_j; \delta_j^*). \end{aligned}$$

Next consider the stage-one rule, where substitution of (4.6) into (B.5) gives

$$\begin{aligned} a_i^*(u_i,x;y_i) &= \sum_{u_{\setminus i}}\sum_{\hat{x}}\left[\sum_{m=1}^{n} c(\hat{x}_m,x)+\lambda c(u_m,x)\right] p(x)p(u_{\setminus i},\hat{x}|x,y_i;\delta_i^*,\gamma_{\setminus i}^*,u_i) \\ &= p(x)\sum_{u_{\setminus i}}\sum_{\hat{x}}\left[\sum_{m=1}^{n} c(\hat{x}_m,x)+\lambda c(u_m,x)\right]\times \\ &\qquad\qquad p(\hat{x}_i|y_i,u_{ne(i)};\delta_i^*)p(u_{\setminus i},\hat{x}_{\setminus i}|x,u_i;\gamma_{\setminus i}^*) \\ &= p(x)\left[\lambda c(u_i,x)+\sum_{\hat{x}_{ne(i)}} p(\hat{x}_{ne(i)}|x,u_i;\gamma_{\setminus i}^*)\sum_{m\in ne(i)} c(\hat{x}_m,x)\right]+ \\ &\quad p(x)\left[\sum_{u_{\setminus i}}\sum_{\hat{x}_{\setminus ne(i)}} p(u_{\setminus i},\hat{x}_{\setminus ne(i)}|x,y_i;\delta_i^*,\gamma_{\setminus i}^*)\left(\sum_{m\notin ne(i)} c(\hat{x}_m,x)+\lambda\sum_{m\neq i} c(u_m,x)\right)\right]. \end{aligned}$$

Only the terms in the first bracket depend upon candidate decision u_i, and all other terms thus have no bearing on the minimization in (4.2). That is, for the stage-one rule in Proposition 4.1, it suffices to satisfy

$$a_i^*(u_i,x;y_i)\propto p(x)\left[\lambda c(u_i,x)+\sum_{j\in ne(i)}\sum_{\hat{x}_j} p(\hat{x}_j|x,u_i;\gamma_{\setminus i}^*)c(\hat{x}_j,x)\right]$$

and, by virtue of (B.1),

$$\begin{aligned} p(\hat{x}_j|x,u_i;\gamma_{\setminus i}^*) &= \sum_{u_{ne(j)\setminus i}} p(u_{ne(j)\setminus i},\hat{x}_j|x,u_i;\gamma_{\setminus i}^*) \\ &= \sum_{u_{ne(j)\setminus i}} p(\hat{x}_j|x,u_{ne(j)};\gamma_j^*)p(u_{ne(j)\setminus i}|x;\gamma_{\setminus i}^*) \\ &= \sum_{u_{ne(j)\setminus i}}\left(\sum_{z_j} p(z_j|x,u_{ne(j)})p(\hat{x}_j|x,z_j;\delta_j^*)\right)\prod_{m\in ne(j)\setminus i} p(u_m|x;\mu_m^*), \end{aligned}$$

where in the last step we have employed the identity $p(\hat{x}_j|x,z_j;\gamma_j^*)=p(\hat{x}_j|x,z_j;\delta_j^*)$ highlighted earlier in the proof. Observe that parameters a_i^* no longer depend upon the local measurement y_i.

Appendix C

On Multi-Stage Communication Architectures: Proofs

■ C.1 Optimal Parameterization of Detection Stage

Proposition 5.1 is proven via analogous steps to those taken in the proof to Proposition 4.1 for the detection rule in the single-stage undirected architecture. Notice that the distribution $p(u_i, \hat{x}_i|x, u_{tr(i)}; \gamma_i)$ in Lemma 5.1 is structurally identical to its counterpart for the single-stage undirected architecture, albeit here both u_i and $u_{tr(i)}$ are discrete-valued length-T vectors.

The final-stage rule δ_i^* minimizes J in (3.1) over all Δ_i, holding the local communication rules and the rules local to all other nodes fixed, if and only if the process $\hat{X}_i = \delta_i(Y_i, I_i^{T+1}, Z_i^{T+1})$ minimizes

$$E\left[c(U, \hat{X}_{\setminus i}, \hat{x}_i, X)|Y_i, I_i^{T+1}, Z_i^{T+1}; \mu_i^*, \gamma_{\setminus i}^*\right], \tag{C.1}$$

over all possible realizations $\hat{x}_i \in \mathcal{X}_i$, with probability one. Fix a realization $\hat{x}_i$ and consider the distribution $p(u, \hat{x}_{\setminus i}, x|y_i, I_i^{T+1}, z_i^{T+1}; \mu_i^*, \gamma_{\setminus i}^*, \hat{x}_i)$ underlying (C.1), or equivalently

$$p(u, \hat{x}_{\setminus i}|x, y_i, I_i^{T+1}, z_i^{T+1}; \mu_i^*, \gamma_{\setminus i}^*, \hat{x}_i)p(x|y_i, I_i^{T+1}, z_i^{T+1}; \mu_i^*, \gamma_{\setminus i}^*, \hat{x}_i).$$

By virtue of Lemma 5.1, for every $(u_i, z_i) \in \mathcal{U}_i \times \mathcal{Z}_i$ such that $p(y_i, I_i^{T+1}, z_i^{T+1}; \mu_i^*, \gamma_{\setminus i}^*) > 0$, the first term simplifies to

$$p(u, \hat{x}_{\setminus i}|x, y_i, I_i^{T+1}, z_i^{T+1}; \mu_i^*, \gamma_{\setminus i}^*) = p(u_i|y_i, I_i^{T+1}; \mu_i^*)p(u_{\setminus i}, \hat{x}_{\setminus i}|x, I_i^{T+1}, z_i^{T+1}; \gamma_{\setminus i}^*)$$

and the second term simplifies to

$$p(x|y_i, I_i^{T+1}, z_i^{T+1}; \mu_i^*, \gamma_{\setminus i}^*) = \frac{p(x)p(I_i^{T+1}, z_i^{T+1}|x; \mu_i^*, \gamma_{\setminus i}^*)p(y_i|x, I_i^{T+1}, z_i^{T+1}; \mu_i^*)}{p(y_i, I_i^{T+1}, z_i^{T+1}; \mu_i^*, \gamma_{\setminus i}^*)}.$$

Taking the product of the two terms, the positive-valued denominator neither depends on x nor on $\hat{x}_i$ and, as such, has no bearing on the minimization of (C.1): moreover,

$$p(u_{\setminus i}, \hat{x}_{\setminus i}|x, I_i^{T+1}, z_i^{T+1}; \gamma_{\setminus i}^*)p(I_i^{T+1}, z_i^{T+1}|x; \mu_i^*, \gamma_{\setminus i}^*) = p(u, z_i, \hat{x}_{\setminus i}|x; \mu_i^*, \gamma_{\setminus i}^*).$$

Now, because $u_i \subset I_i^{T+1}$ and we have already assumed $p(y_i, I_i^{T+1}, z_i^{T+1}; \mu_i^*, \gamma_{\setminus i}^*) > 0$, we have that $p(u_i|y_i, I_i^{T+1}; \mu_i^*) = 1$. Applying Lemma 5.1 and Lemma 5.2 given δ_i is unspecified, we have that

$$p(u_i, z_i|x, u_{tr(i)}; \mu_i^*) = p(z_i|x, u_{tr(i)})p(u_i|x, z_i; \mu_i^*)$$

and

$$p(u_i|x, z_i; \mu_i^*) = \prod_{t=1}^{T} p(u_i^t|x, I_i^t, z_i^t; \mu_i^*),$$

respectively. Similarly, applying Lemma 5.3 with δ_i unspecified, we have that

$$p(y_i|x, u_i, z_i; \mu_i^*) = p(y_i|x, u_i, z_i^1, \ldots, z_i^T; \mu_i^*) = p(y_i|x, I_i^{T+1}; \mu_i^*),$$

which simply states that Z_i^{T+1} is independent (conditioned on X and I_i^{T+1}) of the local measurement process Y_i (recall that $\mathcal{Z}_i^{T+1}$ is empty if $\mathcal{F}$ is directed). Finally, again appealing to Lemma 5.1, we obtain

$$p(u, z_i, \hat{x}_{\setminus i}|x; \mu_i^*, \gamma_{\setminus i}^*) = p(u_i, z_i|x, u_{tr(i)}; \mu_i^*) \prod_{j \neq i} p(u_j, \hat{x}_j|x, u_{tr(j)}; \gamma_j^*).$$

Altogether, it suffices to require that $\hat{X}_i = \delta_i(Y_i, I_i^{T+1}, z_i^{T+1})$ minimize

$$\sum_{x \in \mathcal{X}} b_i^*(\hat{X}_i, x; u_i, z_i)p(Y_i|x, I_i^{T+1}; \mu_i^*)$$

with probability one, where for each candidate decision $\hat{x}_i$,

$$b_i^*(\hat{x}_i, x; u_i, z_i) = p(x) \sum_{u_{\setminus i}} \sum_{\hat{x}_{\setminus i}} c(u, \hat{x}, x)p(u, z_i, \hat{x}_{\setminus i}|x; \mu_i^*, \gamma_{\setminus i}^*). \tag{C.2}$$

■ C.2 Detection-Stage Offline Computation

Starting from Proposition 5.1, we follow essentially the same steps taken in the proof to Proposition 4.2 (and starting from Proposition 4.1) for the detection rule in the single-stage undirected architecture. First note that, with Assumption 5.3 in effect, each ith factor in Lemma 5.1 specializes to

$$\begin{aligned} p(u_i, \hat{x}_i|x, u_{tr(i)}; \gamma_i) &= \sum_{z_i \in \mathcal{Z}_i} p(z_i|x_i, u_{tr(i)}) \int_{y_i \in \mathcal{Y}_i} p(y_i|x_i)p(u_i, \hat{x}_i|y_i, z_i; \gamma_i)\, dy_i \\ &= p(u_i, \hat{x}_i|x_i, u_{tr(i)}; \gamma_i), \end{aligned}$$

leading to detection-stage parameters

$$\beta_i^*(\hat{x}_i, x_i; u_i, z_i) = \sum_{\hat{x}_{\setminus i}} p(x) \sum_{u_{\setminus i}} \sum_{\hat{x}_{\setminus i}} c(u, \hat{x}, x) p(u, z_i, \hat{x}_{\setminus i} | x; \mu_i^*, \gamma_{\setminus i}^*) \tag{C.3}$$

in Proposition 5.2 with underlying probabilistic structure specializing to

$$p(u, z_i, \hat{x}_{\setminus i} | x; \mu_i^*, \gamma_{\setminus i}^*) = p(u_i, z_i | x_i, u_{tr(i)}; \mu_i^*) \prod_{j \neq i} p(u_j, \hat{x}_j | x_j, u_{tr(j)}; \gamma_j^*),$$

$$p(u_i, z_i | x_i, u_{tr(i)}; \mu_i^*) = p(z_i | x_i, u_{tr(i)}) p(u_i | x_i, z_i; \mu_i^*)$$

Now, with Assumption 5.4 in effect, we may substitute (5.8) into (C.3) and observe that the only term in which candidate decision $\hat{x}_i$ appears specializes to

$$c(\hat{x}_i, x_i) \sum_{x_{\setminus i}} p(x) \sum_{u_{\setminus i}} p(u_i, z_i | x_i, u_{tr(i)}; \mu_i^*) \prod_{j \neq i} \sum_{\hat{x}_j} p(u_j, \hat{x}_j | x_j, u_{tr(j)}; \gamma_j^*).$$

Appealing to Lemma 5.3, local to each node $j \neq i$ we have

$$\sum_{\hat{x}_j} p(u_j, \hat{x}_j | x_j, u_{tr(j)}; \gamma_j^*) \quad = \quad p(u_j | x_j, u_{tr(j)}^{1:T}; \mu_j^*)$$

Altogether, it suffices to choose rule parameters

$$\beta_i^*(\hat{x}_i, x_i; u_i, z_i) \propto p(x_i) p(u_i, z_i | x_i; \mu^*) c(\hat{x}_i, x_i)$$

with

$$p(u_i, z_i | x_i; \mu^*) = \sum_{u_{\setminus i}} p(u_i, z_i | x_i, u_{tr(i)}; \mu_i^*) \sum_{x_{\setminus i}} p(x_{\setminus i} | x_i) \prod_{j \neq i} p(u_j | x_j, u_{tr(j)}^{1:T}; \mu_j^*).$$

Bibliography

[1] Saeed A. Aldosari and Jose M. F. Moura. Detection in decentralized sensor networks. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, volume 2, pages 277–280, May 2004.

[2] S. Alhakeem and P. K. Varshney. A unified approach to the design of decentralized detection systems. *IEEE Transactions on Aerospace and Electronic systems*, 31 (1):9–20, January 1995.

[3] S. Alhakeem and P. K. Varshney. Decentralized bayesian detection with feedback. *IEEE Transactions on Systems, Man, and Cybernetics—Part A: Systems and Humans*, 26(4):503–513, July 1996.

[4] Rodney J. Baxter. *Exactly Solved Models in Statistical Mechanics*. Academic Press, New York, NY, 1982.

[5] Dimitri P. Bertsekas. *Dynamic Programming and Optimal Control*, volume 1. Athena Scientific, Belmont, MA, 1995.

[6] Dimitri P. Bertsekas. *Nonlinear Programming*. Athena Scientific, Belmont, MA, 1995.

[7] Dimitri P. Bertsekas, Angelia Nedic, and Asuman E. Ozdaglar. *Convex Analysis and Optimization*. Athena Scientific, Belmont, MA, 2003.

[8] Dimitri P. Bertsekas and John N. Tsitsiklis. *Introduction to Probability*. Athena Scientific, Belmont, MA, 2002.

[9] Dimitri Bertsimas and John N. Tsitsiklis. *Introduction to Linear Optimization*. Athena Scientific, Belmont, MA, 1997.

[10] Vincent D. Blondel and John N. Tsitsiklis. A survey of computational complexity results in systems and control. *Automatica*, 36:1249–1274, 2000.

[11] Rick S. Blum, Saleem A. Kassam, and H. Vincent Poor. Distributed detection with multiple sensors: Part II–Advanced topics. *Proceedings of the IEEE*, 85(1): 64–79, January 1997.

[12] Béla Bollobás. *Modern Graph Theory*. Springer-Verlag, New York, NY, 1998.

[13] Pierre Brémaud. *Markov Chains: Gibbs Fields, Monte Carlo Simulation, and Queues*. Springer-Verlag, New York, NY, 1999.

[14] Olivier Cappé, Eric Moulines, and Tobias Rydén. *Inference in Hidden Markov Models*. Cambridge University Press, Cambridge, UK, 1998.

[15] Jean-Francois Chamberland and Venugopal V. Veeravalli. Decentralized detection in sensor networks. *IEEE Transactions on Signal Processing*, 51(2):407–416, February 2003.

[16] Biao Chen, Ruixiang Jiang, Teerasit Kasetkasem, and Pramod K. Varshney. Channel aware decision fusion in wireless sensor networks. *IEEE Transactions on Signal Processing*, 52(12):3454–3458, December 2004.

[17] Biao Chen and Peter K. Willett. On the optimality of the likelihood-ratio test for local sensor decision rules in the presence of nonideal channels. *IEEE Transactions on Information Theory*, 51(2):693–699, February 2005.

[18] Lei Chen, Martin J. Wainwright, Müjdat Çetin, and Alan S. Willsky. Data association based on optimization in graphical models with application to sensor networks. *Mathematical and Computer Modeling: Special Issue on Optimization and Control for Military Applications*, 43(9-10):1114–1135, May 2006.

[19] Chee-Yee Chong and Srikanta P. Kumar. Sensor networks: Evolution, opportunities, and challenges. *Proceedings of the IEEE: Special Issue on Sensor Networks and Applications*, 91(8):1247–1256, August 2003.

[20] Kenneth C. Chou, Alan S. Willsky, and Albert Benveniste. Multiscale recursive estimation, data fusion, and regularization. *IEEE Transactions on Automatic Control*, 39(3):464–478, March 1994.

[21] Randy Cogill and Sanjay Lall. An approximation algorithm for the discrete team decision problem. *SIAM Journal on Control and Optimization*, 45(4):1359–1368, 2005.

[22] Thomas M. Cover and Joy A. Thomas. *Elements of Information Theory*. John Wiley & Sons, New York, NY, 1991.

[23] Robert G. Cowell, A. Philip Dawid, Steffen L. Lauritzen, and David J. Spiegelhalter. *Probabilistic Networks and Expert Systems*. Springer-Verlag, New York, NY, 1999.

[24] Christopher Crick and Avi Pfeffer. Loopy belief propagation as a basis for communication in sensor networks. In *Proceedings of the 19th Conference on Uncertainty in Artificial Intelligence*. Acapulco, Mexico, August 2003.

[25] Richard Durbin, Sean R. Eddy, Anders Krogh, and Graeme Mitchison. *Biological Sequence Analysis*. Cambridge University Press, Cambridge, UK, 1998.

[26] Leon K. Ekchian. *Optimal Design of Distributed Detection Networks*. PhD thesis, Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology, Cambridge, MA, 1982.

[27] Yariv Ephraim and Neri Merhav. Hidden markov processes. *IEEE Transactions on Information Theory*, 48(6):1518–1569, June 2002.

[28] Brendan J. Frey. *Graphical Models for Machine Learning and Digital Communication*. MIT Press, Cambridge, MA, 1998.

[29] Brendan J. Frey, Ralf Koetter, and Nemanja Petrovic. Very loopy belief propagation for unwrapping phase images. In *Advances in Neural Information Processing Systems 14*, pages 737–743. MIT Press, Cambridge, MA, 2002.

[30] Thomas L. Gabrielle. Information criterion for threshold determination. *IEEE Transactions on Information Theory*, 6:484–486, October 1966.

[31] Robert G. Gallager. *Low-Density Parity Check Codes*. MIT Press, Cambridge, MA, 1963.

[32] Robert G. Gallager. *Information Theory and Reliable Communication*. John Wiley & Sons, 1968.

[33] Robert G. Gallager. *Discrete Stochastic Processes*. Kluwar Academic Publishers, Norwell, MA, 1996.

[34] Andrea J. Goldsmith and Stephen B. Wicker. Design challenges for energy-constrained ad hoc wireless networks. *IEEE Wireless Communications*, pages 8–27, August 2002.

[35] Geoffrey Grimmett. A theorem about random fields. *Bulletin of the London Mathematical Society*, 5:81–84, 1973.

[36] H.R. Hashemi and Ian B. Rhodes. Decentralized sequential detection. *IEEE Transactions on Information Theory*, 35(3):509–520, May 1989.

[37] W. A. Hashlamoun and P. K. Varshney. An approach to the design of distributed bayesian detection structures. *IEEE Transactions on Systems, Man and Cybernetics*, 21(5):1206–1211, September/October 1991.

[38] Carl W. Helstrom. Gradient algorithm for quantization levels in distributed detection systems. *IEEE Transactions on Aerospace and Electronic Systems*, 31(1): 390–398, January 1995.

[39] Tom Heskes. Stable fixed points of loopy belief propagation are minima of the Bethe free energy. In *Advances in Neural Information Processing Systems 15*, pages 359–366, Cambridge, MA, 2003. MIT Press.

[40] Tom Heskes. On the uniqueness of loopy belief propagation fixed points. *Neural Computation*, 16:2379–2413, 2004.

[41] Shan-Yuan Ho. *Distributed Detection and Coding in Information Networks*. PhD thesis, Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology, Cambridge, MA, 2006.

[42] Yu-Chi Ho. Team decision theory and information structures. *Proceedings of the IEEE*, 68:644–654, 1980.

[43] Imad Y. Hoballah and Pramod K. Varshney. Distributed bayesian signal detection. *IEEE Transactions on Information Theory*, 35(5):995–1000, September 1989.

[44] Robert V. Hogg and Allen T. Craig. *Introduction to Mathematical Statistics*. Macmillan, New York, NY, 4th edition, 1978.

[45] Gregory T. Huang. Casting the wireless sensor net. *MIT Technology Review*, pages 50–56, July/August 2003.

[46] P. J. Huber. A robust version of the probability ratio test. *Annals of Mathematical Statistics*, 36:1753–1758, December 1965.

[47] Alexander T. Ihler, John W. Fisher, III, and Alan S. Willsky. Loopy belief propagation: Convergence and effects of message errors. *Journal of Machine Learning Research*, 6(11):905–936, May 2005.

[48] William W. Irving. Problems in decentralized detection. Master's thesis, Massachusetts Institute of Technology, Cambridge, MA, 1991.

[49] Finn V. Jensen and Thomas Nielsen. *Bayesian Networks and Decision Graphs*. Springer Verlag, New York, NY, 2007.

[50] Michael I. Jordan. *Learning in Graphical Models*. The MIT Press, Cambridge, MA, 1999.

[51] Michael I. Jordan. Graphical models. *Statistical Science (Special Issue on Bayesian Statistics)*, 19:140–155, 2004.

[52] Michael I. Jordan, Zoubin Ghahramani, Tommi S. Jaakkola, and Lawrence K. Saul. An introduction to variational methods for graphical models. In Michael I. Jordan, editor, *Learning in Graphical Models*, pages 105–161. The MIT Press, Cambridge, MA, 1999.

[53] Daphne Koller, Uri Lerner, and Dragomir Angelov. A general algorithm for approximate inference and its application to hybrid bayes nets. In Kathryn B. Laskey and Henri Prade, editors, *Proceedings of the 15th International Conference on Uncertainty in Artificial Intelligence*, pages 324–333. Morgan Kaufmann, 1999.

[54] O. Patrick Kreidl and Alan S. Willsky. An efficient message passing algorithm for optimizing decentralized detection networks. In *Proceedings of the 45th IEEE Conference on Decision and Control*, pages 6776–6783, December 2006.

[55] O. Patrick Kreidl and Alan S. Willsky. Inference with minimal communication: a decision-theoretic variational approach. In *Advances in Neural Information Processing Systems*, volume 18. MIT Press, Cambridge, MA, 2006.

[56] O. Patrick Kreidl and Alan S. Willsky. Decentralized detection in undirected network topologies. In *IEEE Statistical Signal Processing Workshop*, Madison, WI, August 2007.

[57] Frank R. Kschischang and Brendan J. Frey. Iterative decoding of compound codes by probability propagation in graphical models. *IEEE Journal on Selected Areas in Communication*, 16(2):219–230, February 1998.

[58] Frank R. Kschischang, Brendan J. Frey, and Hans-Andrea Loeliger. Factor graphs and the sum-product algorithm. *IEEE Transactions on Information Theory*, 47 (2):498–519, February 2001.

[59] Jean-Marc Laferté, Patrick Pérez, and Fabrice Heitz. Discrete Markov image modeling and inference on the quadtree. *IEEE Transactions on Image Processing*, 9(3):390–404, March 2000.

[60] Steffen L. Lauritzen. *Graphical Models*. Oxford University Press, New York, NY, 1996.

[61] Mark R. Luettgen, W. Clem Karl, Alan S. Willsky, and Robert R. Tenney. Multiscale representation of markov random fields. *IEEE Transactions on Signal Processing*, 41(12):3377–3396, December 1993.

[62] Mark R. Luettgen and Alan 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.

[63] Dmitry M. Malioutov, Jason K. Johnson, and Alan S. Willsky. Walk-sum interpretation and analysis of gaussian belief propagation. In *Advances in Neural Information Processing Systems*, volume 18. MIT Press, Cambridge, MA, 2006.

[64] J. Marschak, R. Radner, et al. *The Economic Theory of Teams*. Yale University Press, New Haven, CT, 1972.

[65] Robert J. McEliece, David J. C. MacKay, and Jung-Fu Cheng. Turbo decoding as an instance of pearl's "belief propagation" algorithm. *IEEE Journal on Selected Aeas in Communication*, 16(2):140–152, February 1998.

[66] Thomas P. Minka. Expectation propagation for approximate bayesian inference. In *Proceedings of the 17th International Conference on Uncertainty in Artificial Intelligence*, Seattle, WA, August 2001.

[67] Joris M. Mooij and Hilbert J. Kappen. Sufficient conditions for convergence of the sum-product algorithm. *IEEE Transactions on Information Theory*, 53(12): 4422–4437, December 2007.

[68] James R. Munkres. *Topology*. Prentice Hall, Upper Saddle River, NJ, 2000.

[69] Kevin P. Murphy. The bayes net toolbox for Matlab. *Computing Science and Statistics*, 33, 2001.

[70] Kevin P. Murphy, Yair Weiss, and Michael I. Jordan. Loopy belief propagation for approximate inference: an empirical study. In Kathryn B. Laskey and Henri Prade, editors, *Proceedings of the 15th International Conference on Uncertainty in Artificial Intelligence*, pages 467–475. Morgan Kaufmann, 1999.

[71] James M. Ortega and Werner C. Rheinboldt. *Iterative Solution of Nonlinear Equations in Several Variables*. Academic Press, Inc., London, UK, 1970.

[72] Dimitris A. Pados, Karen W. Halford, Dimitri Kazakos, and P. Papantoni-Kazakos. Distributed binary hypothesis testing with feedback. *IEEE Transactions on Systems, Man, and Cybernetics*, 25(1):21–42, January 1995.

[73] Christos H. Papadimitriou and John N. Tsitsiklis. Intractable problems in control theory. *SIAM Journal on Control and Optimization*, 24:639–654, 1986.

[74] Jason D. Papastavrou and Michael Athans. A distributed hypothesis-testing team decision problem with communications cost. In *Proceedings of the 25th IEEE Conference on Decision and Control*, pages 219–225, December 1986.

[75] Jason D. Papastavrou and Michael Athans. Distributed detection by a large team of sensors in tandem. *IEEE Transactions on Aerospace and Electronic Systems*, 28(3):639–653, July 1992.

[76] Jason D. Papastavrou and Michael Athans. On optimal distributed decision architectures in a hypothesis testing environment. *IEEE Transactions on Automatic Control*, 37(8):1154–1169, August 1992.

[77] Mark A. Paskin and Carlos E. Guestrin. Robust probabilistic inference in distributed systems. In *Proceedings of the 20th Conference on Uncertainty in Artificial Intelligence*. Banff, Canada, July 2004.

[78] Mark A. Paskin, Carlos E. Guestrin, and Jim McFadden. A robust architecture for inference in sensor networks. In *Proceedings of the 4th International Symposium on Information Processing in Sensor Networks*. Los Angelas, CA, April 2005.

[79] Judea Pearl. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufmann, San Mateo, CA, 1988.

[80] A. Pete, K. R. Pattipati, and D. L. Kleinman. Optimization of detection networks with multiple event structures. *IEEE Transactions on Automatic Control*, 39(8): 1702–1707, August 1994.

[81] Andras Pete, Krishna R. Pattipati, and David L. Kleinman. Optimization of decision networks in structured task environments. *IEEE Transactions on Systems, Man, and Cybernetics—Part A: Systems and Humans*, 26(6):739–748, November 1996.

[82] H. Vincent Poor and John B. Thomas. Applications of ali-silvey distance measures in the design of generalized quantizers for binary decision systems. *IEEE Transactions on Communications*, 25(9):893–900, September 1977.

[83] Javed Pothiawala. Analysis of a two-sensor tandem distributed detection network. Master's thesis, Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology, Cambridge, MA, 1989.

[84] Lawrence R. Rabiner. A tutorial on hidden markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2):257–286, February 1989.

[85] Lawrence R. Rabiner and Biing-Hwang Juang. *Fundamentals of Speech Recognition*. Prentice Hall, Englewood Cliffs, NJ, 1993.

[86] R. Radner. Team decision problems. *Annals of Mathematical Statistics*, 33(3): 857–881, September 1962.

[87] Vijay Raghunathan, Curt Schurgers, Sung Park, and Mani B. Srivastava. Energy-aware wireless microsensor networks. *IEEE Signal Processing Magazine*, pages 40–50, March 2002.

[88] Constantino Rago, Peter Willett, and Yaakov Bar-Shalom. Censoring sensors: A low-communication-rate scheme for distributed detection. *IEEE Transactions on Aerospace and Electronic Systems*, 32(2):554–568, April 1996.

[89] Tom Richardson. The geometry of turbo-decoding dynamics. *IEEE Transactions on Information Theory*, 46(1):9–23, January 2000.

[90] Venkatesh Saligrama, Murat Alanyali, and Onur Savas. Distributed detection in sensor networks with packet losses and finite capacity links. *IEEE Transactions in Signal Processing*, 54(11):4118–4132, November 2006.

[91] Sujay Sanghavi, Dmitry M. Malioutov, and Alan S. Willsky. Linear programming analysis of loopy belief propagation for weighted matching. In J.C. Platt, D. Koller, Y. Singer, and S. Roweis, editors, *Advances in Neural Information Processing Systems 20*. MIT Press, Cambridge, MA, 2008.

[92] Akbar M. Sayeed, Deborah Estrin, Gregory G. Pottie, and Kannan Ramchandran. Guest editorial: Self-organizing distributed collaborative sensor networks. *IEEE Journal on Selected Areas in Communications*, 23(4):689–692, April 2005.

[93] Gilbert Strang. *Introduction to Linear Algebra*. Wellesley-Cambridge Press, Wellesley, MA, 1993.

[94] Erik B. Sudderth, Martin J. Wainwright, and Alan S. Willsky. Embedded trees: Estimation of gaussian processes on graphs with cycles. *IEEE Transactions on Signal Processing*, 52(11):3136–3150, November 2004.

[95] Erik. T. Sudderth, Alexander T. Ihler, William T. Freeman, and Alan S. Willsky. Nonparameteric belief propagation. In *Proceedings of the 2003 IEEE Conference on Computer Vision and Pattern Recognition*, volume 1, pages 605–612, June 2003.

[96] Z.B. Tang, K.R. Pattipati, and D.L. Kleinman. Optimization of detection networks: Part I–Tandem structures. *IEEE Transactions on Systems, Man and Cybernetics*, 21(5):1044–1059, September/October 1991.

[97] Z.B. Tang, K.R. Pattipati, and D.L. Kleinman. Optimization of detection networks: Part II–Tree structures. *IEEE Transactions on Systems, Man and Cybernetics*, 23(1):211–221, January/February 1993.

[98] Zhuang-Bo Tang, Krishna R. Pattipati, and David L. Kleinman. An algorithm for determining the decision thresholds in a distributed detection problem. *IEEE Transactions on Systems, Man and Cybernetics*, 21(1):231–237, January/February 1991.

[99] Sekhar Tatikonda and Michael I. Jordan. Loopy belief propogation and gibbs measures. In *Proceedings of the 18th Annual Conference on Uncertainty in Artificial Intelligence (UAI-02)*, pages 493–500, San Francisco, CA, 2002. Morgan Kaufmann Publishers.

[100] Sekhar Tatikonda and Michael I. Jordan. Loopy belief propogation and gibbs measures. In *Proceedings of the 40th Allerton Conference on Communication, Control and Computing*, Monticello, IL, October 2002.

[101] Wee-Peng Tay, John N. Tsitsiklis, and Moe Z. Win. Asymptotic performance of a censoring sensor network. *IEEE Transactions on Information Theory*, 53(11): 4191–4209, November 2007.

[102] Wee-Peng Tay, John N. Tsitsiklis, and Moe Z. Win. Data fusion trees for detection: Does architecture matter? 2007. To appear.

[103] Demosthenis Teneketzis and Yu-Chi Ho. The decentralized wald problem. *Information and Computation*, 73:23–44, April 1987.

[104] Robert R. Tenney and Nils R. Sandell, Jr. Detection with distributed sensors. *IEEE Transactions on Aerospace Electronic Systems*, 17(4):501–510, September 1981.

[105] John N. Tsitsiklis. Decentralized detection by a large number of sensors. *Mathematics of Control, Signals and Systems*, 1:167–182, 1988.

[106] John N. Tsitsiklis. Decentralized detection. In H. Vincent Poor and John B. Thomas, editors, *Advances in Statistical Signal Processing*, volume 2, pages 297–344. JAI Press, Greenwich, CT, 1993.

[107] John N. Tsitsiklis and Michael Athans. On the complexity of decentralized decision making and detection problems. *IEEE Transactions on Automatic Control*, AC-30(5):440–446, May 1985.

[108] Harry L. Van Trees. *Detection, Estimation, and Modulation Theory*, volume 1. John Wiley & Sons, New York, NY, 1968.

[109] Pramod K. Varshney. *Distributed Detection and Data Fusion*. Springer-Verlag, New York, NY, 1997.

[110] Ramanarayanan Viswanathan and Pramod K. Varshney. Distributed detection with multiple sensors: Part I–Fundamentals. *Proceedings of the IEEE*, 85(1): 54–63, January 1997.

[111] Martin J. Wainwright, Tommi S. Jaakkola, and Alan 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.

[112] Martin J. Wainwright, Tommi S. Jaakkola, and Alan S. Willsky. Tree-reweighted belief propagation algorithms and approximate ML estimation by pseudo-moment matching. In *Ninth International Workshop on Artificial Intelliegnce and Statistics*. Key West, FA, January 2003.

[113] Martin J. Wainwright and Michael I. Jordan. Graphical models, exponential families and variational inference. Technical Report 649, UC-Berkeley Department of Statistics, Berkeley, CA, September 2003.

[114] Yair Weiss. Correctness of local probability propagation in graphical models with loops. *Neural Computation*, 12(1):1–41, January 2000.

[115] Max Welling, Thomas P. Minka, and Yee Whye Teh. Structured region graphs: Morphing EP into GBP. In *Proceedings of the 21st Conference in Uncertainty in Artificial Intelligence*, volume 21, pages 609–614. Edinburgh, Scotland, July 2005.

[116] Max Welling and Yee Whye Teh. Approximate inference in boltzmann machines. *Artificial Intelligence*, 143(1):19–50, January 2003.

[117] Joe Whittaker. *Graphical Models in Applied Multivariate Statistics*. John Wiley & Sons, New York, NY, 1990.

[118] Wim Wiegerinck. Approximations with reweighted generalized belief propagation. In *Tenth International Workshop on Artificial Intelliegnce and Statistics*. Barbados, January 2005.

[119] Peter Willett, Peter F. Swaszek, and Rick S. Blum. The good, bad, and ugly: Distributed detection of a known signal in dependent gaussian noise. *IEEE Transactions on Signal Processing*, 48(12):3266–3279, December 2000.

[120] Alan S. Willsky. Multiresolution markov models for signal and image processing. *Proceedings of the IEEE*, 90(8):1396–1458, August 2002.

[121] Alan S. Willsky et al. Distributed fusion in sensor networks: a graphical models perspective. In Ananthram Swami, Qing Zhao, Yao-Win Hong, and Lang Tong, editors, *Wireless Sensor Networks: Signal Processing and Communications*, pages 215–249. John Wiley & Sons, New York, NY, 2007.

[122] Jin-Jun Xiao and Zhi-Quan Luo. Universal decentralized detection in a bandwidth constrained sensor network. *IEEE Transactions on Signal Processing*, 53(8):2617–2624, August 2005.

[123] Jonathan S. Yedidia, William T. Freeman, and Yair Weiss. Constructing free-energy approximations and generalized belief propagation algorithms. *IEEE Transactions on Information Theory*, 51(7):2282–2312, July 2005.

[124] Feng Zhao, Jie Liu, Juan Liu, Leonidas Guibas, and James Reich. Collaborative signal and information processing: an information-directed approach. *Proceedings of the IEEE: Special Issue on Sensor Networks and Applications*, 91(8):1199–1209, August 2003.

[125] Yunmin Zhu, Rick S. Blum, Zhi-Quan Luo, and Kon Max Wong. Unexpected properties and optimum-distributed sensor detectors for dependent observation cases. *IEEE Transactions on Automatic Control*, 45(1):62–72, January 2000.