Bayesian inference of chemical reaction networks

by

Nikhil Galagali

B.Tech., Indian Institute of Technology Madras (2007)
S.M., Massachusetts Institute of Technology (2009)

Submitted to the Department of Mechanical Engineering
in partial fulfillment of the requirements for the degree of

Doctor of Philosophy in Computational Science and Engineering

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

June 2016

Author . . . .
Department of Mechanical Engineering
February 19, 2016

Certified by . . . .
Youssef M. Marzouk
Associate Professor of Aeronautics and Astronautics
Thesis Supervisor

Accepted by . . . .
Rohan Abeyaratne
Chairman, Department Committee on Graduate Students

Bayesian inference of chemical reaction networks

by

Nikhil Galagali

Submitted to the Department of Mechanical Engineering
on February 19, 2016, in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy in Computational Science and Engineering

Abstract

The development of chemical reaction models aids system design and optimization, along with fundamental understanding, in areas including combustion, catalysis, electrochemistry, and biology. A systematic approach to building reaction network models uses available data not only to estimate unknown parameters, but to also learn the model structure. Bayesian inference provides a natural approach for this data-driven construction of models.

Traditional Bayesian model inference methodology is based on evaluating a multidimensional integral for each model. This approach is often infeasible for reaction network inference, as the number of plausible models can be very large. An alternative approach based on model-space sampling can enable large-scale network inference, but its efficient implementation presents many challenges. In this thesis, we present new computational methods that make large-scale nonlinear network inference tractable.

Firstly, we exploit the network-based interactions of species to design improved "between-model" proposals for Markov chain Monte Carlo (MCMC). We then introduce a sensitivity-based determination of move types which, when combined with the network-aware proposals, yields further sampling efficiency. These algorithms are tested on example problems with up to 1000 plausible models. We find that our new algorithms yield significant gains in sampling performance, with almost two orders of magnitude reduction in the variance of posterior estimates.

We also show that by casting network inference as a fixed-dimensional problem with point-mass priors, we can adapt existing adaptive MCMC methods for network inference. We apply this novel framework to the inference of reaction models for catalytic reforming of methane from a set of ≈ 32000 possible models and real experimental data. We find that the use of adaptive MCMC makes large-scale inference of reaction networks feasible without the often extensive manual tuning that is required with conventional approaches.

Finally, we present an approximation-based method that allows sampling over very large model spaces whose exploration remains prohibitively expensive with exact sampling methods. We run an MCMC algorithm over model indicators and for each visited model approximate the model evidence via Laplace's method. Limited and sparse available data tend to produce multi-modal posteriors over the model indicators. To perform inference in this setting, we develop a population-based approximate model inference MCMC algorithm. Numerical tests on problems with around 10^9 models demonstrate the superiority of our population-based algorithm over single-chain MCMC approaches.

Thesis Supervisor: Youssef M. Marzouk
Title: Associate Professor of Aeronautics and Astronautics

Acknowledgments

I want to firstly thank my advisor, Youssef Marzouk, for being a great mentor. His enthusiasm for doing research and the rigor he displays in all his pursuits have been a constant source of inspiration to me. This thesis has been shaped by his many inputs and I am grateful for his guidance. I would also like to thank Ahmed Ghoniem, William Green, Pierre Lermusiaux, and Habib Najm for serving on my thesis committee. Their questions during the committee meetings made me delve deeper into many of the ideas developed in this thesis.

I also would like to acknowledge the members of the Uncertainty Quantification Lab for sharing a congenial atmosphere to work in. Thank you also to all the friends I have made during stay at MIT. Spending time with friends has definitely made the graduate school experience more enjoyable.

I am grateful for the generous financial support from KAUST Global Research Partnership and BP through the BP-MIT Energy Conversion Program during my graduate studies. I am also thankful to Ms. Leslie Regan for the administrative support.

Finally, I would like to thank my parents and my brother for their love and support over all the years I have spent away from home.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Motivation

Detailed chemical reaction networks are a critical component of simulation tools in a wide range of applications, including combustion, catalysis, electrochemistry, and biology. In addition to being used as predictive tools, network models are also key to developing an improved understanding of the complex process under study. The development of reaction network models typically entails three tasks: selection of participating species, identification of species interactions (refered to as reactions) and the calibration of unknown parameter values. Combustion chemistry has a rich history of building reaction models. Large reaction models with sometimes thousands of reactions are well known [28, 67]. In other areas such as systems biology, catalysis, and electrochemistry, the construction of network models can frequently be extremely challenging due to the limited understanding of the operating reaction pathways. For instance, there exist a number of competing hypotheses about H_2 and CO oxidation mechanisms for a solid-oxide fuel cell [62]. Reconstruction of biological networks involved in cell signalling, gene regulation and metabolism is one of the major challenges in systems biology due to the specificity of species interactions

[2, 17, 44, 65]. A standard approach to building models in such a case is to postulate reaction networks and to compare them based on their ability to reproduce indirect system-level experimental data. Data-driven approaches to network learning involve defining a metric of fit, e.g., penalized least-squares, cross-validation, model evidence, etc. and selecting models that optimize this metric. As such, the development of models involves not only the identification of the right model structure, but also the estimation of underlying parameter values given available data.

Bayesian model inference provides a rigorous statistical framework for fusing data with prior knowledge to yield a full description of model and parameter uncertainties [13, 46, 111]. The application of Bayesian model inference to reaction networks, however, presents a significant computational challenge. Model discrimination in Bayesian analysis is based on computing model probabilities conditioned on available data, i.e., *posterior* model probabilities. Formally, the posterior model probability of a model M_n is given by

$$p(M_n|\boldsymbol{\mathcal{D}}) = \frac{p(M_n)p(\boldsymbol{\mathcal{D}}|M_n)}{\sum_n p(M_n)p(\boldsymbol{\mathcal{D}}|M_n)},$$

where

$$p(\mathcal{D}|M_n) = \int \cdots \int p(\mathcal{D}|\boldsymbol{k_n}, M_n)p(\boldsymbol{k_n}|M_n)d\boldsymbol{k_n}$$

is known as the model evidence, $\boldsymbol{k_n}$ is the parameter vector of model M_n, and $\boldsymbol{\mathcal{D}}$ refers to the available data. An approach to Bayesian model inference is to assume that the relationship between species is described by linear or discrete functionals and model parameters take conjugate priors, thereby making the calculation of individual posterior model probabilities analytically tractable. It is, however, often widely believed that species interactions are more appropriately defined by the law of mass action. The law of mass action gives the rate of a chemical reaction (say $X + Y \rightarrow Z$) as the product of a reaction-specific rate constant k with reactant concentrations $[X]$

and $[Y]$:

$$\text{Rate} = -k[X][Y]. \tag{1.1}$$

Under quasi-steady-state assumptions, the law of mass action produces reaction rate expression for enzymatic reactions that are known as Michaelis-Menten functionals [91]. The reaction rate for an enzyme E binding to a substrate S to produce product P ($E + S \rightarrow E + P$) by Michaelis Menten kinetics is given by

$$\text{Rate} = k_{cat}[E]_0 \frac{[S]}{k_M + [S]}, \tag{1.2}$$

where k_{cat} denotes the rate constant, $[E]_0$ is the enzyme concentration, $[S]$ the substrate concentration, and k_M the Michaelis constant. Using the law of mass action to define reaction rate produces a system of differential equations such that the parameter-to-observable map (*forward model*) is typically nonlinear. These equations can be further embedded into differential equation models that describe convective and diffusive transport, surface interactions, and other physical phenomena that affect experimental observations. Rigorous computation of posterior model probabilities then requires evaluation of a high-dimensional integral for each model. A number of sampling-based methods exist in the literature for this purpose [26, 47, 94], but they are computationally taxing. When the number of competing models becomes large, the above methods actually become computationally *infeasible*.

Reaction network inference is particularly prone to this difficulty, since the number of plausible models can grow exponentially with the number of proposed reactions. A systematic approach to network inference requires appraising a combinatorially large number of models: instead of a few model hypotheses, one might start with a list of proposed reactions, for example, and form a collection of plausible models by considering all valid combinations of the proposed reactions. Alternatives such as Laplace's method and Bayesian information criterion have been suggested [21, 82, 109], but they involve approximations of the posterior distribution. Across-model sampling offers a solution in cases where the number of models is large [24, 57, 73]. These methods work by making the sampler jump between models to explore the joint posterior distribution over models and parameters. Model probabilities are estimated from the number of times the sampler visits each model. The prohibitively high cost of model comparisons based on the computation of evidence for each model is avoided as the sampler visits each model in proportion to its posterior probability. Efficient across-model sampling, however, is challenging and require a delicate design of proposals for between-model moves. Many practical applications of across-model sampling methods have relied on pilot posterior explorative runs to get a rough idea of the posterior distribution, although a few automated methods do exist in literature [56, 110]. The effective use of across-model sampling methods continues to be a challenge, especially for problems where the parameter-to-observable map is nonlinear.

1.2 Background on reaction network inference

The general problem of network inference has been tackled in the past with various modeling choices and inferential approaches. The modeling of species interactions has spanned from simple Boolean networks to detailed physics-based differential equations and stochastic models. The Boolean network approach relies on simple ON/OFF switches and standard logic interactions to describe species interactions. Additive linear or generalized linear models take an intermediate approach in terms of complexity and reliability. The differential equations based network interactions are at the other end of the complexity spectrum, but being rooted in mechanistic models hold the promise of better understanding and improved predictions. From an algorithmic standpoint, a large number of inference methods, both from a Bayesian and a non-Bayesian standpoint, have been proposed. We review here a few network inference approaches and refer the readers to some detailed reviews for the complete story [44, 96, 98].

1.2.1 Non-Bayesian reaction network inference

Many non-Bayesian methods for network inference have been published in the chemical and biological engineering literature. Gardner et al. adopted an ODE formulation and developed a technique known as Network Identification by Multiple Regression (NIR) [43]. Their approach constructs a first-order model of regulatory interactions and uses multiple linear regression to infer species interactions. Bansal et al. developed an algorithm known as the Time Series Network Identification in which they assumed a linear ODE model for species interactions and inferred the network topology by a combination of interpolation and principal component analysis [10]. Bonneau et al. use L1 shrinkage to identify transcriptional influences on genes based on the integration of genome annotation and expression data [18]. Margolin et al. have proposed another technique called ARCANE that adopts an information-theoretic approach in which they identify candidate interactions by estimating pairwise species mutual information [84]. Nachman et al. utilize dynamic Bayesian network models and the structural EM algorithm for network identification [92]. Another technique, correlation metric construction, suggested by Arkin et al. is based on the calculation and analysis of a time-lagged multivariate correlation function of a set of time-series of chemical concentrations [7]. Burnham et al. propose a statistical technique relying on t-statistics and R^2 for the inference of chemical reaction networks governed by ordinary differential equations [22].

1.2.2 Bayesian reaction network inference

The Bayesian approach to network inference has seen increasing applications in areas such as protein signalling modeling, gene regulation reconstruction, combustion chemistry etc. The inference methods for signalling topologies and gene regulation pathways have principally been developed with linear or discrete formulations [40, 90, 107, 116]. Using Gaussian or multinomial likelihood functions and conjugate priors with these formulations leads to model evidence being available in closed form.

In spite of the cheap analytical evaluation of evidence, the exponential explosion of the number of networks given species and their possible interactions precludes direct enumeration of model evidence. Thus, sampling based approaches have been developed for large-scale network inference in such contexts [37, 39]. At the same time, ODE-based species interaction models (mass-action kinetics) are also being incorporated into inference frameworks. The use of ODE-based forward models oftentimes results in nonlinear parameter-observable dependency—network inference then has to be based on the computation of model evidence numerically. Xu et al. applied Bayesian model inference with nonlinear ODEs for the elucidation of ERK signalling pathway [118]. Braman et al. used Bayesian methodology for the comparison of syngas chemistry models [19]. However, the methods used above for numerical computation of evidence are limited to applications with a small number of hand-crafted models. Large-scale network inference with nonlinear forward models has seen very limited work. Oates et al. applied Bayesian model selection for the comparison of systematically generated models derived from ODE-based species interactions [95]. They used reversible-jump Markov chain Monte Carlo algorithm, a general across-model sampling method, for the simultaneous sampling of network topologies and their underlying parameters. As discussed in 1.1, the use of vanilla across-model sampling methods are generally known to perform poorly. There is a need for the development of efficient large-scale network inference methods that would allow a systematic comparison of exponentially large number of networks, but one that incorporates nonlinear forward models emerging from ODE-based species interaction formulations.

1.3 Thesis contributions

In this thesis, we present methods for efficient large-scale Bayesian inference of nonlinear chemical reaction networks. We develop algorithms that exploit structural properties of chemical reaction networks to improve exploration of posteriors over a large number of reaction networks in comparison to existing methods. Further, we develop a model-space sampling approach that makes approximations of the parameter posterior, and thereby allows sampling over very large model spaces whose exploration remains intractable with exact sampling methods.

More specifically, we operate in the across-model sampling framework and make four contributions:

1. **Network inference with adaptive MCMC**

   The rate of a chemical reaction is given by the law of mass action and the net species production rate a species is given by the species production rate from all reactions [77]. The species production rates further feed into forward models that describe convective and diffusive transport, surface interactions, and other phenomena affecting experimental obsevations. Nevertheless, the additive structure of the net species production rate means that reaction inclusion/exclusion can be controlled by setting the rate constants to non-zero/zero values. In spite of the overall nonlinear dependency of the observables on the rate constants. This indirect control of network topology by assigning specific values to the rate constants means that the plausible networks are statistically nested. Nested models provide a natural between-model move construction. Nested models can further allow the use of *fixed-dimensional* Markov chain Monte Carlo (MCMC) algorithms. Adaptive MCMC in which the parameter proposals adapt to posterior samples are known for fixed-dimensional MCMC algorithms and improve sampling efficiency without manual tuning. We exploit the nested structure of reaction network problems and develop an adaptive MCMC algorithm for network inference. The developed algorithm is used to learn reaction networks for steam and dry catalytic reforming of methane on rhodium from a set of 15 proposed reactions and real experimental data.

2. **Network-aware sampling**

Chemical reaction networks can quickly become very complex [28, 108]. The network of species interactions, however, has a special structure hidden in it. The production/destruction of a species is directly linked to the concentration of other species it is participating in a reaction with. Therefore, the rate of production/destruction of a species will necessarily be zero if those other species are absent from the system. From a data-analytic perspective, the available data cannot inform the presence of reactions with zero reaction-rate. Many species in a reaction network such as catalysts or enzymes play a fundamental role in the operation of reactions, but do not get consumed. Moreover, practically feasible experiments yield data that is sparse—data is only linked to a few of the species. Sparsity of data and presence of catalyst/enzymes can further render some reactions ineffective in influencing the observables. Thus, the inclusion of these reactions is also not informed by data. We develop a network-aware across-modeling sampling algorithm that recognizes the *effective networks* being inferred and exploits this knowledge to design efficient parameter proposals for moves between models. This translates into superior sampling performance and low-variance posterior estimates. The recognition of effective networks also allows derandomization of some conditional expectations and thereby yields further variance reduction.

3. **Sensitivity-based network-aware sampling**

   Not all reactions in a network are equally important in influencing the observables. The identification of reactions that have a sharper impact on observables can be critical in designing improved across-model samplers. We develop a local sensitivity-based metric to identify *key* reactions and use this to develop better between-model move proposals for the reversible jump MCMC algorithm. Combining the sensitivity-based proposal construction along with network-aware sampling produces a highly improved nonlinear network inference algorithm. We apply the algorithm for the inference of network topology from a set of 1024 systematically generated networks that were obtained from a subset of proposed reactions for the activation of the extracellular signal-regulated kinase pathway by epidermal growth factor [118].

4. **Network inference with approximation**

   The network inference methods in Contributions 1, 2, and 3 are exact sampling methods, i.e., they are guaranteed to converge to the correct posterior distribution asymptotically as sampling proceeds. Exact inference methods over the joint space of models and parameters are essential for consistency of posterior estimates and their development is an important goal. However, for very large model spaces, exact sampling may still be very expensive. We develop an approximation-based network inference approach by using Laplace's method to approximate model evidences and using Markov chain Monte Carlo to explore the posterior distribution *only* over model indicators. Nonlinearity of the forward model and limited available data results in the posterior distribution over models being multimodal. To explore multimodal posterior distributions, we extend the approximate posterior inference to a population-based network inference algorithm. The developed algorithm is then used to infer signalling networks from a space of 10^9 plausible networks.

This thesis is organized into 6 chapters. Following the introduction in Chapter 1, Chapter 2 gives a detailed overview of the motivations for model inference, contrasts different model inference approaches, highlights Bayesian model inference, and discusses existing numerical methods for Bayesian model inference. Chapters 3, 4, and 5 present the four main contributions of this thesis. We summarize and discuss future work in Chapter 6.

Chapter 2

Model inference: formulation and numerical approaches

An integral component of scientific research is the construction of models for the physical process under study. Models are created for two main reasons: they enable an easy understanding of a complex process by breaking it down into more readily interpretable modules, and models can be used for making predictions of unobserved quantities. Development of reliable models is often very hard because one may only get to observe a few noisy realizations of the physical process—referred to as *data*. Utilizing the available data and any background information about the process, the job of a model developer is to construct a consistent set of equations that relate the model inputs and model parameters to the quantities of interest. Historically, models have been built by empiricism and experimental investigation. More recently, first-principles calculations have also been used to aid model development in disciplines such as chemistry and biology. However, the development of faster computers and the availability of high-quality data has now allowed the use of rigorous statistical techniques in the model development phase. Given a set of plausible models and some data, tools from statistical inference can be used for a systematic evaluation of all models to identify the "best" set of models. In this chapter, we discuss the motivations for model inference, present a popular philosophy for effective learning of models from data, outline common approaches for model inference, introduce the Bayesian paradigm for model learning, and discuss some numerical techniques for Bayesian model inference.

2.1 Model inference

Model inference can be defined informally as the assessment of models to ascertain the degree to which each is supported by available data. A prerequisite for model inference is the availability of (i) plausible models and (ii) relevant data to discriminate among the models. Very often, we also have a great deal of background information about the quality of competing models and the values of their underlying parameters. This knowledge—termed as *prior information*—can be incorporated in the model inference framework. It is important at this stage to distinguish *model inference* from the common practice of *model reduction* in chemical kinetics [14, 97]. Model reduction refers to a systematic reduction in the size of a large kinetic model so as to reproduce model outputs within a specified tolerance. Such a procedure, however, assumes that an accurate model (i.e., the full kinetic model) is already known and fixed. And, crucially, it does not take experimental data into account during reduction.

A model of a physical process describes a specific collection of input-output relationships. In particular, a model describes how some pre-specified *quantities of interest* are related to input variables. As a result, a model may preclude the description of quantities for which it has not been specifically built. Figure 2-1 shows a typical process model. This model—consisting of governing equations expressing conservation laws, reaction network models, and thermo-kinetic parameters—may relate inputs such as concentration C_{in}, temperature T_{in}, pressure P, and applied voltage ΔV to observables such as concentration C_{out}, ignition delay τ_{ign}, and current I.

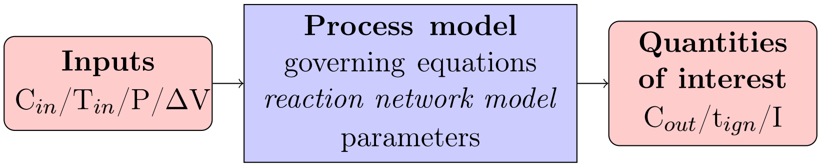


Figure 2-1: A process model

2.2 Goals of model inference

To be able to prescribe a set of rules that determine the best or the most likely model, we need to precisely define the purpose of model inference. Model inference is performed primarily for two purposes: interpretation and prediction. Frequently, the models being compared during inference are physics-based. In such cases, the selected model can be used to gain valuable insight into the operating mechanism. This kind of insight is often used for experimental design. The other main objective of model inference is to make predictions of the quantities of interest.

The selection of the best model based on available data and prior information is essentially a statistical problem. As the amount of data grows, the precision of inference improves and our confidence in the selected model grows. However, in most practical situations, the amount of data required to strongly discriminate between models is unavailable. This engenders significant uncertainty in the inferred results. Thus it is imperative—for reliable inference and accurate quantification of prediction uncertainties—that the model inference method provide means to quantify model uncertainty.

In the next section, we discuss commonly used criteria for model inference and highlight Occam's razor, which is a powerful approach for model discrimination based on the intuitive idea of balancing model fit with complexity.

2.3 Approaches for model inference

Having presented a few motivations for model inference, we now proceed to discuss three approaches for model choice. Since the data we would be using to infer the best model will necessarily be noisy, it would be incorrect to try to fit exactly to all available data. If we maximize the quality of fit to the available data, it is the most complex model—model with the largest degrees of freedom—which typically would best fit the data. As we discuss in the following paragraphs, such a strategy would be sub-optimal since the objective is to select models that peform well for all data, not just the observed data.

2.3.1 Model selection based on estimation of prediction error

Model inference is sometimes performed in a data rich situation. In such settings, the available data can be used to compute an estimate of prediction error known as the *empirical prediction error* [64]. To begin with, the available data is split into three parts: a training set, a validation set, and a test set. The training set is used to fit the models; the validation set is used to estimate the prediction error for model selection; the test set is used for the assessment of the generalization error of the final chosen model. In a slightly data deficient situation, test set can be used to select the model as well as estimate the prediction error. In such cases, the final chosen model will necessarily underestimate the prediction error [64].

2.3.2 Cross validation

Cross-validation is another method that is used often to estimate the prediction error. A K-fold cross-validation procedure begins by splitting the available data into K sets. Then a model is trained using data from $K-1$ sets as the training data and the prediction error computed on the K^{th} set as the test set. This process is repeated for all K sets and then the average prediction error computed. By performing these operations for all M competing models, one can select the most well supported model or rank the competing models. Mathematically, we let $s : \{1, ..., N\}| \rightarrow \{1, ..., K\}$ be an indexing function that indicates the partition to which data point n is allotted. We denote by $\hat{f}^{-s}(x)$ the fitted function, computed with the s^{th} part of the data removed. Then the cross-validation estimate of the prediction error is

$$CV = \frac{1}{N} \sum_{n=1}^{N} L(y_n, \hat{f}^{-s(n)}(x_n)) \tag{2.1}$$

When $K = N$, the cross-validation method is referred to as *leave-one-out* cross validation [64].

2.3.3 Goodness-of-fit and complexity penalty

A well known observation pertaining to the fitting of model parameters to available data is that the goodness-of-fit generally improves as the model complexity grows. Though the mismatch between model predictions and available data decreases as the model complexity increases, we would not expect our future predictions to be very accurate. This is because by increasing model complexity we begin to fit to the noise in the data. This problem is known as the problem of *overfitting* in statistics. Thus, a common strategy is to adopt a model inference criterion such that the mismatch between model predictions and available data is agreeable and at the same time model complexity is limited. This two-fold objective is also described by the bias-variance tradeoff [64]. As the complexity of the model grows, the variance of the inferred parameters would be high, and as a result, the expected prediction error of the model tends to be high. On the contrary, simpler models tend to have higher bias, and so fit poorly to available data and have high prediction error (Figure 2-2).

The best models tend to be ones that balance bias with variance. Therefore, a popular approach to model selection is one that rewards good agreement with available data, but also penalizes model complexity. This guiding principle for the

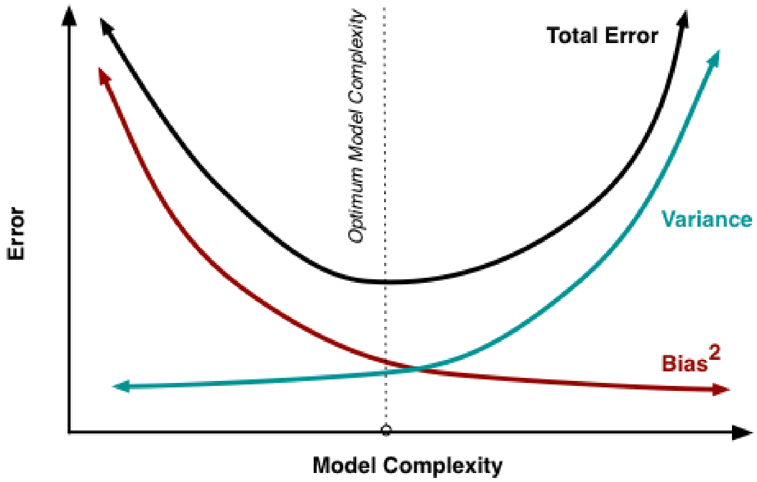


Figure 2-2: Bias-variance tradeoff curve

assessment of models, first suggested by William of Ockham, is encapsulated by the *Occam's razor* [82]. The Occam's razor principle is suitable irrespective of whether the objective model inference is interpretation or prediction. The explanation for prediction is clear from the discussion from the last paragraph. Even in cases where the goal of model inference is interpretation, it makes sense that we determine model strength not just based on fit to available data. In this thesis, we work with methods that balance bias with variance. In such cases, the key is to determine the penalty term that would penalize model complexity appropriately.

2.4 Balancing goodness-of-fit with model complexity

The goal of this section is to discuss methods that balance the quality of fit to available data with the complexity of the fitted model. The cross-validation (CV) method presented in the last section is most suitable when the amount of available data is plentiful [8, 64]; in a data-poor context, the cross-validation metric is noisy and its results highly variable. A method that explicitly incorporates an Occam's razor is more useful for the data-deficient case one typically encounters in reaction network inference.

In statistics in general, there are two main viewpoints for the identification of likely models and their underlying parameter values from data. The *frequentist* approach to learning treats the models and parameters as fixed unknown quantities that are determined by techniques that aim to produce good estimation over all possible data sets. The *Bayesian* approach, in contrast, regards the models and parameters as random variables whose distributions conditioned on available data are determined by the consistent application of the rules of probability theory. Model selection approaches in the frequentist setting, such as C_p-statistic, Akaike information criterion, etc, impose an Occam's razor by selecting models based on the following optimization problem:

$$M^* = \arg\min_M ||\boldsymbol{\mathcal{D}} - \boldsymbol{G}_M(\boldsymbol{k}_M)|| + \alpha |M|,$$

where M^* is the optimal model, $\boldsymbol{G}_M$ is the prediction with model M, $\boldsymbol{k}_M$ are the parameters of model M, $\boldsymbol{\mathcal{D}}$ the observed data, $||\boldsymbol{\mathcal{D}} - \boldsymbol{G}_M(\boldsymbol{k}_M)||$ is the data misfit, $|M|$ is the model complexity, and α is the penalty on model complexity. The problem with the above optimization based approaches is that they tend to be *ad hoc*, due to a lack of a clear guideline about the right value for the penalty α [12].

2.5 Bayesian approach to model inference

Bayesian statistics provides a rigorous inference framework to assimilate noisy and indirect data, a natural mechanism to incorporate prior knowledge from different sources, and a full description of uncertainties in parameter values and model structure. It is based on Bayes' rule of probability:

$$p(\boldsymbol{k}|\boldsymbol{\mathcal{D}}) = \frac{p(\boldsymbol{\mathcal{D}}|\boldsymbol{k})p(\boldsymbol{k})}{p(\boldsymbol{\mathcal{D}})} \tag{2.2}$$

Here, $\boldsymbol{k}$ is the parameter being inferred, $p(\boldsymbol{k}|\boldsymbol{\mathcal{D}})$ is the posterior probability density of $\boldsymbol{k}$ conditioned on data $\boldsymbol{\mathcal{D}}$, $p(\boldsymbol{\mathcal{D}}|\boldsymbol{k})$ is the likelihood of observing $\boldsymbol{\mathcal{D}}$ given the parameter value, and $p(\boldsymbol{k})$ is the prior probability density of parameter $\boldsymbol{k}$. $p(\boldsymbol{\mathcal{D}})$, commonly refered to as *evidence* or *marginal likelihood*, is the marginal distribution of data. Sampling the posterior enables description of posterior uncertainty and the estimation of posterior summaries such as the mean and standard deviation. Posterior exploration by sampling is seldom directly feasible except for conjugate prior distributions. For nonlinear forward models and/or non-conjugate prior distributions, one has to rely on an indirect sampling approach, such as importance sampling or Markov chain Monte Carlo [3, 51]. Application of Bayesian parameter inference to physical models has received much recent interest [9, 74, 89, 113], with applications ranging from geophysics [35, 55] and climate modeling [71] to reaction kinetics [53, 78, 99].

Applying Bayes' rule to models M, we get

$$p(M|\boldsymbol{\mathcal{D}}) = \frac{p(\boldsymbol{\mathcal{D}}|M)p(M)}{p(\boldsymbol{\mathcal{D}})}. \tag{2.3}$$

Comparing the posterior of any two models, M_i and M_j, yields the posterior odds:

$$\frac{p(M_i|\boldsymbol{\mathcal{D}})}{p(M_j|\boldsymbol{\mathcal{D}})} = \frac{p(\boldsymbol{\mathcal{D}}|M_i)p(M_i)}{p(\boldsymbol{\mathcal{D}}|M_j)p(M_j)} \tag{2.4}$$

Assuming that all models are equally probable before the observation of data, we get

$$\frac{p(M_i|\boldsymbol{\mathcal{D}})}{p(M_j|\boldsymbol{\mathcal{D}})} = \frac{p(\boldsymbol{\mathcal{D}}|M_i)}{p(\boldsymbol{\mathcal{D}}|M_j)}. \tag{2.5}$$

The quantity on the right-hand side of Equation 2.5 is known as *Bayes factor* and is the traditional metric used to compare the probabilities of different models [13, 46, 75]. A key advantage of this Bayesian formulation is an implicit penalty on model complexity in the model evidence—an automatic Occam's razor that guards against overfitting [82]. Computation of Bayes factor, however, is expensive as it relies on the evaluation of high-dimensional integrals. Specifically,

$$\frac{p(\boldsymbol{\mathcal{D}}|M_i)}{p(\boldsymbol{\mathcal{D}}|M_j)} = \frac{\int p(\boldsymbol{\mathcal{D}}|\boldsymbol{k_i})p(\boldsymbol{k_i}|M_i)d\boldsymbol{k_i}}{\int p(\boldsymbol{\mathcal{D}}|\boldsymbol{k_j})p(\boldsymbol{k_j}|M_j)d\boldsymbol{k_j}} \tag{2.6}$$

where $\boldsymbol{k_i}$ and $\boldsymbol{k_j}$ are model-specific multidimensional parameters. Alternatives such as Laplace approximation method and Bayesian information criterion have been suggested in the literature [21, 82, 109], but they all involve making approximations about distributions. The standard approach presented above becomes infeasible for an exhaustive comparison of a large number of models because of the high computational cost involved. As mentioned in Chapter 1, in this thesis, we focus on developing tractable network inference methodologies when the number of plausible models is large. The underlying rate parameter uncertainties would come “for free" as a natural byproduct of the model inference results.

The Bayesian posterior model probabilities also have the advantage of being easily interpretable. Having computed the posterior probabilities of the models, it is straightforward to understand the degree to which the different models are supported by available data based on their posterior probabilities. The Bayesian model inference procedure has another favorable property in that it is consistent. Consistency is a property that if the true model is among the set of models being compared, then the posterior probability of the true model converges to 1 in probability as the size of the data set goes to infinity.

2.6 Numerical methods for Bayesian computation

The generation of samples from posterior distributions is a central problem in Bayesian statistics. Integration of functions with respect to posterior distributions in high-dimensions is most efficiently performed by Monte Carlo sampling. Formally, the posterior distribution can be defined over a general state space

$$\boldsymbol{\Theta} \in \bigcup_{M} \{M\} \times \boldsymbol{k}_M, \tag{2.7}$$

where $\boldsymbol{k}_M \subseteq \mathbb{R}^M$ are parameter spaces and each parameter space $\boldsymbol{k}_M$ could have a different dimensionality. M here acts as an indicator of the individual parameter spaces. The posterior distribution over $\boldsymbol{\Theta}$ is again given by Bayes' rule:

$$p(M, \boldsymbol{k}_M|\boldsymbol{\mathcal{D}}) = \frac{p(\boldsymbol{\mathcal{D}}|\boldsymbol{k}_M, M)p(M)p(\boldsymbol{k}_M|M)}{\sum_M \int_{\boldsymbol{k}_M} p(\boldsymbol{\mathcal{D}}|\boldsymbol{k}_M, M)p(M)p(\boldsymbol{k}_M|M)}. \tag{2.8}$$

Note, in relation to models and parameters discussed in the previous section, M would correspond to model indicators and $\boldsymbol{k}_M$ then are their respective parameter vectors. The posterior distribution over M and $\boldsymbol{k}_M$ are related through a marginalization:

$$p(M|\boldsymbol{\mathcal{D}}) = \int_{\boldsymbol{k}_M} p(M, \boldsymbol{k}_M|\boldsymbol{\mathcal{D}}) \tag{2.9}$$

When the forward model is nonlinear or the prior distributions are non-conjugate, the sampling of the posterior distribution $p(\boldsymbol{\Theta}|\boldsymbol{\mathcal{D}})$ using standard Monte Carlo is infeasible. In such cases, one has to resort to advanced Monte Carlo methods, the most widely useful of which are a general class of algorithms known as the Markov chain Monte Carlo methods.

In the following sections, we discuss various Monte Carlo methods that enable computation of posterior model probabilities and in many cases also produce samples from parameter posterior distributions. We begin with a brief discussion of an algorithm used for fixed-dimensional posterior sampling known as the Metropolis-Hastings algorithm.

2.6.1 Metropolis-Hastings algorithm

In many problems of interest, the model structure is assumed to be well known. Thus the target of the inference procedure is only the posterior distribution over the underlying parameters of the model. Sampling in such a fixed-dimensional setting, when direct sampling is infeasible, is commonly performed using the Metropolis-Hastings algorithm. The Metropolis-Hastings algorithm is an iterative algorithm that produces a Markov chain whose limiting distribution is the posterior distribution $p(\boldsymbol{k}|\boldsymbol{\mathcal{D}})$. At each step of the algorithm, a sample from a distribution $q(\boldsymbol{k}|\boldsymbol{k^n})$ known as the proposal distribution is generated. The proposed sample is accepted as the new state of the chain with a probability that depends on the posterior and proposal distributions. If the proposed sample is rejected, the current state of the chain is retained as the new state. The steps of the Metropolis-Hastings algorithm are given in Algorithm 1.

Algorithm 1 The Metropolis-Hastings algorithm

```
Given: Data $\boldsymbol{\mathcal{D}}$, prior density $p(\boldsymbol{k})$, likelihood function $p(\boldsymbol{\mathcal{D}}|\boldsymbol{k})$, proposal density
  $q(\boldsymbol{k}|\boldsymbol{k^n})$, number of steps $N$
Initialize $\boldsymbol{k}^0$
for $n = 0$ to $N - 1$ do
    Sample $u \sim \mathcal{U}_{[0,1]}$
    Sample $\boldsymbol{k^*} \sim q(\boldsymbol{k}|\boldsymbol{k^t})$
    if $u < \alpha(\boldsymbol{k^t}, \boldsymbol{k^*}) = \min\left\{1, \frac{p(\boldsymbol{k^*}|\boldsymbol{\mathcal{D}})q(\boldsymbol{k^t}|\boldsymbol{k^*})}{p(\boldsymbol{k^t}|\boldsymbol{\mathcal{D}})q(\boldsymbol{k^*}|\boldsymbol{k^t})}\right\}$ then
        $\boldsymbol{k^{n+1}} = \boldsymbol{k^*}$
    else
        $\boldsymbol{k^{n+1}} = \boldsymbol{k^n}$
    end if
end for
```

Under mild conditions, the above algorithm guarantees convergence of the distribution of the Markov chain state $\boldsymbol{k^n}$ to the posterior distribution and the existence of limit theorems [105]. The sequence of Markov chain iterates yield estimates

$$\bar{f} = \frac{1}{N} \sum_{n=1}^{N} f(\boldsymbol{k^n}). \tag{2.10}$$

that converges almost surely to $\int f(\boldsymbol{k}) p(\boldsymbol{k}|\boldsymbol{\mathcal{D}}) d\boldsymbol{k}$ by the strong law of large numbers. In contrast to standard Monte Carlo sampling, the sequence of iterates produced by the Metropolis-Hastings algorithm are correlated and thus the posterior estimates have a higher variance. Special cases of the general Metropolis-Hastings algorithm are obtained by considering specific choices of the proposal distribution $q(\boldsymbol{k}|\boldsymbol{k^n})$. If the proposal is independent of the current location of the chain, we get the Independence Metropolis-Hastings algorithm. Another very popular algorithm is obtained by considering proposals that consist of independent perturbations about the current state. Specifically, the proposal is of the form $\boldsymbol{k^*} = \boldsymbol{k^n} + \boldsymbol{\epsilon_n}$, where $\boldsymbol{\epsilon_n} \sim q$ is independent of $\boldsymbol{k^n}$. The resulting algorithm is then known as the random-walk Metropolis-Hastings algorithm.

2.6.2 Computing the posterior model probabilities

The evaluation of posterior model probabilities in the Bayesian framework is often a challenging computational problem. The computation of model evidence is seldom analytically tractable. When the forward model is linear, the use of conjugate priors permits closed form solutions for model evidence. However, in many practical situations, we encounter a nonlinear parameter-observable dependency. The evidence is then obtained by resorting to numerical techniques. Formally, the objective is to approximate the posterior model probability distribution $p(M|\boldsymbol{\mathcal{D}})$. For any model M_i,

$$p(M_i|\boldsymbol{\mathcal{D}}) \propto p(\boldsymbol{\mathcal{D}}|M_i) p(M_i), \tag{2.11}$$

where

$$p(\boldsymbol{\mathcal{D}}|M_i) = \int p(\boldsymbol{\mathcal{D}}|\boldsymbol{k_i}, M_i)p(\boldsymbol{k_i}|M_i)d\boldsymbol{k_i}, \tag{2.12}$$

and $p(M_i)$ is the prior probability of model M_i and $\boldsymbol{k_i}$ is the vector of unknown paramters in model M_i. The computation of the above multidimensional integral is carried out by numerical methods. In low-dimensional settings, it is sometimes efficient to compute the integral by numerical quadrature schemes [29, 48]. But for moderate to high-dimensional integrals, sampling-based methods are necessary. We provide a brief overview of some of the commonly used sampling-based methods for Bayesian model inference.

2.6.3 Computing the evidence via model-specific Monte Carlo simulations

Many existing methods in literature compute posterior model probabilities by estimating the model evidence (2.12) individually for all competing models.

Standard Monte Carlo and importance sampling

A simple approach to estimating the model evidence for a model M_i is to evaluate the Monte Carlo sum by sampling from the prior distribution $p(\boldsymbol{k_i})$. The estimate

$$\hat{p}(\boldsymbol{\mathcal{D}}|M_i) = \frac{1}{N}\sum_{n=1}^{N} p(\boldsymbol{\mathcal{D}}|\boldsymbol{k_i^n}), \tag{2.13}$$

where $\boldsymbol{k_i^n} \sim p(\boldsymbol{k_i})$, is guaranteed to converge almost surely to the true model evidence by the strong law of large numbers [105]. Although very simple, this approach is highly inefficient as most samples are drawn from regions of the parameter space where the likelihood tends to have a low value. Practically, this manifests into high variance in evidence estimates.

An improvement to the above estimator is to use importance sampling. Importance sampling involves generating samples from a different distribution $q(\boldsymbol{k_i})$ known as the proposal. Under some general conditions, a simulation consistent estimate is given by

$$\hat{p}(\boldsymbol{\mathcal{D}}|M_i) = \frac{1}{N}\sum_{n=1}^{N} w_n p(\boldsymbol{\mathcal{D}}|\boldsymbol{k_i^n}) \tag{2.14}$$

where $w_n = p(\boldsymbol{k_i^n})/q(\boldsymbol{k_i^n})$ and $\boldsymbol{k_i^n} \sim q(\boldsymbol{k_i})$ [49]. The precision of importance sampling estimates hinges on $q(\boldsymbol{k_i})$ being a good approximation of $p(\boldsymbol{k_i}|\boldsymbol{\mathcal{D}})$ and thus good proposal distributions $q(\boldsymbol{k_i})$ are a priori hard to design in complex multi-dimensional settings.

Posterior harmonic mean estimator

Newton et al. [94] suggested another importance sampling estimator for the estimation of model evidence. In contrast to the standard importance sampling estimator (2.14), an alternative simulation consistent importance sampling estimator is given by

$$\hat{p}(\boldsymbol{\mathcal{D}}|M_i) = \frac{\sum_{n=1}^{N} w_n p(\boldsymbol{\mathcal{D}}|\boldsymbol{k_i^n})}{\sum_{n=1}^{N} w_n}. \tag{2.15}$$

Here again $w_n = p(\boldsymbol{k_i^n})/q(\boldsymbol{k_i^n})$ and $\boldsymbol{k_i^n} \sim q(\boldsymbol{k_i})$ [49]. The advantage of this estimator is that the proposal need only be known upto an unknown constant. Newton et al. [94] noted that the posterior distribution $p(\boldsymbol{k_i}|\boldsymbol{\mathcal{D}})$ is an efficient proposal distribution and if we simulate samples approximately from the posterior, substitution into (2.15) yields an estimate of $p(\boldsymbol{\mathcal{D}}|M_i)$,

$$\hat{p}(\boldsymbol{\mathcal{D}}|M_i) = \left\{\frac{1}{N}\sum_{n=1}^{N} p(\boldsymbol{\mathcal{D}}|\boldsymbol{k_i^n})^{-1}\right\}^{-1}, \tag{2.16}$$

called the harmonic mean estimator. The simulation of posterior samples may be performed by Markov chain Monte Carlo or sequential-importance-resampling methods [3]. It can easily be verified that the estimator (2.16) converges almost surely to the correct model evidence. The drawback of the harmonic estimator is that it can be unstable because $p(\boldsymbol{\mathcal{D}}|\boldsymbol{k_i})^{-1}$ is often not square integrable with respect to the posterior distribution.

Marginal likelihood from the Gibbs and the Metropolis-Hastings output

Another set of approaches proposed by Chib [25] and Chib et al. [26] involves expanding the model evidence in terms of the likelihood, prior, and the posterior density at a parameter value $\boldsymbol{k_i^*}$ and then estimating the posterior density value at $\boldsymbol{k_i^*}$ using samples generated from the posterior distribution. The evidence being the normalizing constant is given by

$$p(\boldsymbol{\mathcal{D}}|M_i) = \frac{p(\boldsymbol{\mathcal{D}}|\boldsymbol{k_i^*})p(\boldsymbol{k_i^*})}{p(\boldsymbol{k_i^*}|\boldsymbol{\mathcal{D}})}. \tag{2.17}$$

If $\hat{p}(\boldsymbol{k_i^*}|\boldsymbol{\mathcal{D}})$ is a posterior estimate, then the estimate of model evidence on the logarithm scale is

$$\log \hat{p}(\boldsymbol{\mathcal{D}}|M_i) = \log p(\boldsymbol{\mathcal{D}}|\boldsymbol{k_i^*}) + \log \hat{p}(\boldsymbol{k_i^*}) - \log \hat{p}(\boldsymbol{k_i^*}|\boldsymbol{\mathcal{D}}). \tag{2.18}$$

When the posterior conditionals $p(\boldsymbol{k_i}|\boldsymbol{\mathcal{D}}, z)$ and $p(z|\boldsymbol{\mathcal{D}}, \boldsymbol{k_i})$ are available, Chib [25] propose using the output from the Gibbs sampler $\{\boldsymbol{k_i^n}, \boldsymbol{z^n}\}_{n=1}^N$ to obtain a Monte Carlo estimate of $p(\boldsymbol{k_i}|\boldsymbol{\mathcal{D}}) = \int p(\boldsymbol{k_i}|\boldsymbol{\mathcal{D}}, \boldsymbol{z})p(\boldsymbol{z}|\boldsymbol{\mathcal{D}})d\boldsymbol{z}$ given as

$$\hat{p}(\boldsymbol{k_i^*}|\boldsymbol{\mathcal{D}}) = N^{-1}\sum_{n=1}^{N} p(\boldsymbol{k^*}|\boldsymbol{\mathcal{D}}, \boldsymbol{z^n}). \tag{2.19}$$

Chib et al. [26] extend the method to cases when full conditionals are intractable and posterior samples are simulated using the Metropolis-Hastings algorithm. If $\{\boldsymbol{k_i^n}\}_{n=1}^N$ are samples from the posterior $p(\boldsymbol{k_i}|\boldsymbol{\mathcal{D}})$ and $\{\boldsymbol{k_i^m}\}_{m=1}^M$ samples from the proposal $q(\boldsymbol{k_i}|\boldsymbol{k_i^*}, \boldsymbol{\mathcal{D}})$, a simulation-consistent estimate of the posterior density is

$$\hat{p}(\boldsymbol{k_i^*}|\boldsymbol{\mathcal{D}}) = \frac{N^{-1}\sum_{n=1}^{N} \alpha(\boldsymbol{k_i^*}|\boldsymbol{k_i^n}, \boldsymbol{\mathcal{D}})q(\boldsymbol{k_i^*}|\boldsymbol{k_i^n})}{M^{-1}\sum_{m=1}^{M} \alpha(\boldsymbol{k_i^m}|\boldsymbol{k_i^*}, \boldsymbol{\mathcal{D}})}. \tag{2.20}$$

Here

$$\alpha(\boldsymbol{k}_i'|\boldsymbol{k}_i,\boldsymbol{\mathcal{D}}) = \min\left\{1, \frac{p(\boldsymbol{\mathcal{D}}|\boldsymbol{k}_i')p(\boldsymbol{k}_i')q(\boldsymbol{k}_i|\boldsymbol{k}_i',\boldsymbol{\mathcal{D}})}{p(\boldsymbol{\mathcal{D}}|\boldsymbol{k}_i)p(\boldsymbol{k}_i)q(\boldsymbol{k}_i'|\boldsymbol{k}_i,\boldsymbol{\mathcal{D}})}\right\} \tag{2.21}$$

is the Metropolis-Hastings acceptance probability.

Path sampling

Methods that generalize the importance sampling algorithm by introducing a sequence of intermediate distributions between two densities whose normalizing constants are to be determined have existing in the computational physics literature for a few decades. The acceptance ratio method and thermodynamic integration are routinely used in statistical physics to compute free energies differences. More recently, Meng et al. [87] and Gelman et al. [47] reinterpret the acceptance ratio method as an instance of bridge sampling and more generally bridge sampling and thermodynamic integration as instances of the path sampling algorithm. Recall that model evidence can be written using Bayes' rule as

$$p(\boldsymbol{\mathcal{D}}|M_i) = \frac{p(\boldsymbol{\mathcal{D}}|\boldsymbol{k}_i)p(\boldsymbol{k}_i)}{p(\boldsymbol{k}_i|\boldsymbol{\mathcal{D}})} \tag{2.22}$$

More generally, the normalizing constant $z(\theta)$ of an unnormalized density $q(\boldsymbol{k}_i|\theta)$ may be written as

$$z(\theta) = \frac{q(\boldsymbol{k}_i|\theta)}{p(\boldsymbol{k}_i|\theta)}, \tag{2.23}$$

where $p(\boldsymbol{k}_i|\theta)$ is a probability density function. Taking logarithms and then differentiating both sides of (2.23) with respect to θ,

$$\frac{d}{d\theta}\log z(\theta) = \int \frac{1}{z(\theta)}\frac{d}{d\theta}q(\boldsymbol{k}_i|\theta)\mu(d\boldsymbol{k}_i) \tag{2.24}$$

$$= \mathbb{E}_{\theta}\left[\frac{d}{d\theta}\log q(\boldsymbol{k_i}|\theta)\right], \tag{2.25}$$

where $\mathbb{E}_{\theta}$ denotes the expectation with respect to $p(\boldsymbol{k_i}|\theta)$.

Let

$$U(\boldsymbol{k_i}, \theta) = \frac{d}{d\theta}\log q(\boldsymbol{k_i}|\theta). \tag{2.26}$$

Integrating (2.25) from 0 to 1 yields

$$\lambda = \log\left[\frac{z(1)}{z(0)}\right] = \int_0^1 \mathbb{E}_{\theta}[U(\boldsymbol{k_i}, \theta)]d\theta. \tag{2.27}$$

If we consider θ as a random variable with a uniform distribution, the right hand side of (2.27) can be considered as the expectation of $U(\boldsymbol{k_i}, \theta)$ over the joint distribution of $(\boldsymbol{k_i}, \theta)$. More generally, introducing a prior density $p(\theta)$ for $\theta \in [0, 1]$ we get

$$\lambda = \mathbb{E}\left[\frac{U(\boldsymbol{k_i}, \theta)}{p(\theta)}\right], \tag{2.28}$$

where the expectation is with respect to the joint density $p(\boldsymbol{k_i}|\theta)p(\theta)$. Identity (2.27) immediately suggests an unbiased estimator of λ:

$$\hat{\lambda} = \frac{1}{N}\sum_{n=1}^{N}\frac{U(\boldsymbol{k_i^n}, \theta^n)}{p(\theta^n)} \tag{2.29}$$

using n draws $(\boldsymbol{k_i^n}, \theta^n)$ from $p(\boldsymbol{k_i}, \theta)$. The choice of the prior density $p(\theta)$ and the number of discretizations of θ detemine the particular variant of importance sampling algorithm. Bridge sampling involves a single intermediate distribution, whereas the path sampling or thermodynmic integration involve a continuous discretization of θ.

Another method which fits into the path sampling framework utilizes powers of the posterior densities in (2.25) to yield formulas for the model evidence that make use of MCMC sampling and numerical integration [41].

Annealed importance sampling

Neal [93] has presented another importance sampling based technique for the computation of model evidence called the annealed importance sampling. The method relies on using an importance proposal over a multidimensional state space with the aid of Markov chain transition kernels with specific invariant distribution. Firstly, a series of tempered posterior distributions

$$f^l(\boldsymbol{k_i}) = f(\boldsymbol{k_i}|\boldsymbol{\mathcal{D}})^{\beta_l} f(\boldsymbol{k_i})^{\beta_l - 1}, \tag{2.30}$$

where $1 = \beta_0 > \beta_1 > ... > \beta_n = 0$, $f(\boldsymbol{k_i}|\boldsymbol{\mathcal{D}})$ is the unnormalized posterior probability density and $f(\boldsymbol{k_i})$ is the prior probability density of $\boldsymbol{k_i}$, are defined. The algorithm starts by generating a sample $\boldsymbol{k_i^n}$ from $f(\boldsymbol{k_i})$. Thereafter, starting from $f^{n-1}(\boldsymbol{k_i})$ samples $\boldsymbol{k_i^l}$ are drawn from $f^l(\boldsymbol{k_i})$ with a Markov kernel $T^l(\boldsymbol{k_i}|\boldsymbol{k_i^{l+1}})$ that keeps $f^l(\boldsymbol{k_i})$ invariant. These Markov kernels are constructed in the usual Metropolis-Hastings or Gibss sampling fashion such that the detailed balance condition is satisfied. This process is repeated J times to generate sequence of n-dimensional samples. Let

$$w_j = \frac{f_{n-1}(\boldsymbol{k_i^{n-1}})}{f_n(\boldsymbol{k_i^{n-1}})} \frac{f_{n-2}(\boldsymbol{k_i^{n-2}})}{f_{n-1}(\boldsymbol{k_i^{n-2}})} ... \frac{f_1(\boldsymbol{k_i^1})}{f_2(\boldsymbol{k_i^1})} \frac{f_0(\boldsymbol{k_i^0})}{f_1(\boldsymbol{k_i^0})} \tag{2.31}$$

be importance weights. The average $\sum w_j/N$ converges to the model evidence.The efficiency of the algorithm increases with the number of tempered distributions.

2.6.4 Across-model Markov chain Monte Carlo

The use of the above model-specific Monte Carlo methods becomes practically infeasible when the number of possible models is large. Common examples include variable selection problems, autoregressive time series modelling, and network inference. In such cases, Monte Carlo methods that simultaneously traverse the space of models and parameters are most favourable. These across-model sampling methods work by making the MCMC sampler jump between models to explore the joint space of models and parameters. Model probabilities are estimated from the number of times the sampler visits each model. The prohibitively high cost of model comparisons based on the computation of evidence for each model is avoided as the sampler visits each model in proportion to its posterior probability. The challenge in the use of across-model sampling schemes, however, is that the design of efficient model-switching proposal distributions can often be hard. This thesis focuses on the across-model sampling framework and we provide here a brief background on existing methodologies.

Product space approach

Carlin et al. [24] introduced an across-model MCMC algorithm by transforming the transdimensional problem into one that is of constant dimension. The central idea is that they assume complete independence of parameter vectors $\{\boldsymbol{k_j}\}_{j=1}^{M}$ given the model indicator M_i and choose 'pseudopriors' $p(\boldsymbol{k_j}|M_{i\neq j})$. From the conditional independence assumptions, the joint distribution of data $\boldsymbol{\mathcal{D}}$ and $\{\boldsymbol{k_j}\}_{j=1}^{M}$ when the model is M_i is

$$p(\boldsymbol{\mathcal{D}}, \{\boldsymbol{k_j}\}_{j=1}^{M}, M_i) = p(\boldsymbol{\mathcal{D}}|\boldsymbol{k_i}, M_i)\bigg\{\prod_{j=1}^{M} p(\boldsymbol{k_j}|M_i)\bigg\}p(M_i) \tag{2.32}$$

Assumming all full conditional distributions given by

$$p(\boldsymbol{k_j}|\boldsymbol{k_{k\neq j}}, M, \boldsymbol{\mathcal{D}}) = \begin{cases} p(\boldsymbol{\mathcal{D}}|\boldsymbol{k_j}, M_j)p(\boldsymbol{k_j}|M_j), & M = M_j \\ p(\boldsymbol{k_j}|M \neq M_j), & M \neq M_j \end{cases} \tag{2.33}$$

can be sampled and

$$p(M_j|\{\boldsymbol{k_j}\}_{j=1}^{M}) = \frac{p(\boldsymbol{\mathcal{D}}|\boldsymbol{k_i}, M_i)\bigg\{\prod_{j=1}^{M} p(\boldsymbol{k_j}|M_i)\bigg\}p(M_i)}{\sum_{k=1}^{M} p(\boldsymbol{\mathcal{D}}|\boldsymbol{k_i}, M_i)\bigg\{\prod_{j=1}^{M} p(\boldsymbol{k_j}|M_i)\bigg\}p(M_i)} \tag{2.34}$$

a Gibbs sampler can be used to generate samples from the joint posterior distribution $p(M, \{\boldsymbol{k_j}\}_{j=1}^{M})$. Specifically,

$$\hat{p}(M_j|\boldsymbol{\mathcal{D}}) = \frac{\text{number of } M_j^n}{\text{total number of samples}},\ \text{j=1,...,M} \tag{2.35}$$

gives simulation-consistent estimates of posterior model probabilities. The drawback of the above method is that it requires simulation from pseudopriors at each iteration and as such the choice of pseudopriors has a direct impact on the simulation efficiency. Carlin et al. [24] note that a good pseudoprior $p(\boldsymbol{k_j}|M_{i\neq j})$ is the conditional posterior distribution $p(\boldsymbol{k_i}|M_i)$. Dellaportas et al. [30] proposed a 'Metropolised' version of the above approach, altering the model selection step into first proposing a move to a model and then accepting the move with Metropolis-Hasings acceptance probability.

Reversible jump Markov chain Monte Carlo

Reversible jump MCMC (RJMCMC) is a general framework for posterior exploration when the dimension of the state space is not constant [56, 57]. Consider the space of candidate models $\mathcal{M} = \{M_1, M_2, ..., M_N\}$. Each model M_j has an n_j-dimensional vector of unknown parameters $\boldsymbol{k}_{M_j} \in \mathcal{R}^{n_j}$, where n_j can different values for different models. The reversible jump MCMC algorithm simulates a Markov chain whose invariant distribution is the joint model-parameter posterior distribution $P(M, \boldsymbol{k}_M|\mathcal{D})$. Each step of the algorithm consists of proposing a new vector of model-parameter values and accepting the proposed values according to an acceptance probability that also depends on the current model-parameter value vector. At any point of the state space, many different proposal moves can be constructed. Generally, the moves can be classified as between-model and within-model moves. The within model move involves using the Metropolis-Hastings proposal. A between model move involves proposing a move to a different model and the corresponding set of parameter values. The necessary conditions for the reversible jump MCMC to be ergodic with the posterior distribution as the invariant distribution is that the transition kernel resulting from the chosen proposal is irreducible and aperiodic [57]. Further, ensuring that the posterior distribution over the models and parameters is the invariant distribution of the Markov chain is accomplished by satisfying the detailed balance condition. The detailed balance condition is enforced by constructing moves between any two models M and M' according to a bijective map $\boldsymbol{f}$ from $(\boldsymbol{k}_M, \boldsymbol{u})$ to $(\boldsymbol{k}_{M'}, \boldsymbol{u'})$, where $\boldsymbol{k}_M$ and $\boldsymbol{k}_{M'}$ are parameters of models M and M', $\boldsymbol{u}$ and $\boldsymbol{u'}$ known as dimension matching variables are such that $\dim(\boldsymbol{k}_{M'}) + \dim(\boldsymbol{u'}) = \dim(\boldsymbol{k}_M) + \dim(\boldsymbol{u})$ and have densities $q(\boldsymbol{u})$ and $q(\boldsymbol{u'})$, respectively, and $\boldsymbol{f}$ and $\boldsymbol{f^{-1}}$ are differentiable (i.e., $\boldsymbol{f}$ is a diffeomorphism). The choice of the distribution of $\boldsymbol{u}$ is part of the proposal construction and in addition to an appropriate $\boldsymbol{f}$ is key to an efficient reversible jump MCMC simulation. At each step of the simulation, given the current state $(M, \boldsymbol{k}_M)$ a move to a new model M' is first proposed according to a chosen distribution $q(M'|M)$. Next, to move to $(M', \boldsymbol{k}_{M'})$ from $(M, \boldsymbol{k}_M)$ involves generating a sample of $\boldsymbol{u}$ according to $q(\boldsymbol{u}|\boldsymbol{k}_M)$ and accepting the proposed move with probability:

$$\alpha(\boldsymbol{k}_M, \boldsymbol{k}_{M'}) = \min\{1, A\}, \tag{2.36}$$

where

$$A = \frac{p(M', \boldsymbol{k}_{M'}|\boldsymbol{\mathcal{D}})q(M|M')q(\boldsymbol{u'}|\boldsymbol{k}_{M'})}{p(M, \boldsymbol{k}_M|\boldsymbol{\mathcal{D}})q(M'|M)q(\boldsymbol{u}|\boldsymbol{k}_M)} \left|\det(\nabla \boldsymbol{f}(\boldsymbol{k}_M, \boldsymbol{u}))\right|, \tag{2.37}$$

and $(\boldsymbol{k}_{M'}, \boldsymbol{u'}) = \boldsymbol{f}(\boldsymbol{k}_M, \boldsymbol{u})$. The reverse move from $(\boldsymbol{k}_{M'}, \boldsymbol{u'})$ to $(\boldsymbol{k}_M, \boldsymbol{u})$ is performed according to $\boldsymbol{f^{-1}}$ and has an acceptance probability $\min\{1, A^{-1}\}$. The complete reversible jump MCMC algorithm we use is given in Algorithm 2.

The selection of a good map $\boldsymbol{f}$ and the design of proposal distribution $q(\boldsymbol{u}|\boldsymbol{k}_M)$ is challenging and often chosen based on pilot runs of the reversible-jump MCMC. The high cost and typically poor performance of the pilot-runs based RJMCMC has prompted the development of methods for automatic proposal construction [1, 20, 36, 38, 59, 58]. All the above methods attempt to increase the acceptance rate of between-model moves at the cost of some additional computational expense and have shown to improve performance in a number of cases.

Algorithm 2 Reversible jump MCMC

```
Given: A set of models $M \in \mathcal{M}$ with corresponding parameter vectors $\boldsymbol{k}_M$, posterior densities $p(M, \boldsymbol{k}_M|\boldsymbol{\mathcal{D}})$, proposal for between-model move $q(M'|M)$, proposal density $q_{M \to M'}(\boldsymbol{u}|\boldsymbol{k}_M)$, jump-function $\boldsymbol{f}$, and Metropolis-Hastings proposal $q(\boldsymbol{k'}_M|\boldsymbol{k}_M)$.
$\alpha \in (0, 1)$: probability of within-model move
Initlialize starting point $(M^0, \boldsymbol{k}_{M^0})$
for $n = 0$ to $N_{iter}$ do
    Sample $b \sim \mathcal{U}_{[0,1]}$
    if $b \leq \alpha$ then
        Sample $p \sim \mathcal{U}_{[0,1]}$
        if $p \leq A(\boldsymbol{k}^{\boldsymbol{n}}_{M^n} \to \boldsymbol{k}^{'}_{M^n}) = \min\left\{1, \frac{p(M^n, \boldsymbol{k'}_{M^n}|\boldsymbol{\mathcal{D}})q(\boldsymbol{k}^{\boldsymbol{n}}_{M^n}|\boldsymbol{k}^{'}_{M^n})}{p(M^n, \boldsymbol{k}^{\boldsymbol{n}}_{M^n}|\boldsymbol{\mathcal{D}})q(\boldsymbol{k}'_{M^n}|\boldsymbol{k}^{n}_{M^n})}\right\}$ then
            $(M^{n+1}, \boldsymbol{k}^{n+1}_{M^{n+1}}) = (M^{'}, \boldsymbol{k}^{'}_{M'})$
        else
            $(M^{n+1}, \boldsymbol{k}^{n+1}_{M^{n+1}}) = (M^n, \boldsymbol{k}^{n}_{M^n})$
        end if
    else
        Sample $M' \sim q(M'|M^n)$
        Sample $\boldsymbol{u} \sim q_{M^n \to M'}(\boldsymbol{u}|\boldsymbol{k}_{M^n})$
        Sample $p \sim \mathcal{U}_{[0,1]}$
        if $p \leq A = \min\left\{1, \frac{p(M', \boldsymbol{k}_{M'}|\boldsymbol{\mathcal{D}})q(M^n|M')q(\boldsymbol{u'}|\boldsymbol{k}_{M'})}{p(M^n, \boldsymbol{k}_{M^n}|\boldsymbol{\mathcal{D}})q(M'|M^n)q(\boldsymbol{u}|\boldsymbol{k}_{M^n})} \left|\frac{\partial \boldsymbol{f}(\boldsymbol{k}_{M^n}, \boldsymbol{u})}{\partial(\boldsymbol{k}_{M^n}, \boldsymbol{u})}\right|\right\}$ then
            $(M^{n+1}, \boldsymbol{k}^{\boldsymbol{n+1}}_{M^{n+1}}) = (M', \boldsymbol{k}_{M'})$
        else
            $(M^{n+1}, \boldsymbol{k}^{\boldsymbol{n+1}}_{M^{n+1}}) = (M^n, \boldsymbol{k}^{\boldsymbol{n}}_{M^n})$
        end if
    end if
end for
```

Other across-model samplers

Godsill [52] introduced a composite framework that allows interpretation of the product space approach and the reversible jump sampler as special cases of the same general approach. Recall that in the product space approach of Carlin et al. [24], sampling is performed over a space $\{\mathcal{M}, \{\boldsymbol{k_j}\}_{j=1}^{M})\}$ of fixed dimension. The posterior distribution $p(M, \boldsymbol{k}|\boldsymbol{\mathcal{D}})$ can be expressed as

$$p(M_i, \boldsymbol{k}|\boldsymbol{\mathcal{D}}) = \frac{p(\boldsymbol{\mathcal{D}}|M_i, \boldsymbol{k_j})p(\boldsymbol{k_j}|M_j)p(\boldsymbol{k_{-j}}|\boldsymbol{k_i}, M_i)p(M_i)}{p(\boldsymbol{\mathcal{D}}|M_i)}. \tag{2.38}$$

Sampling over the fixed dimensional space $\{M, \boldsymbol{k}\}$ in a Metropolis-Hastings framework using appropriate pseudopriors $p(\boldsymbol{k_{-j}}|\boldsymbol{k_i}, M_i)$ and proposal distribution $q(\boldsymbol{k'}|\boldsymbol{k})$, Godsill et al. obtain the product space approach and the reversible jump sampler as special cases.

Before the development of the reversible jump algorithm, Grenander et al. [60] introduced a sampling method known as jump diffusion that involved between-model jumps and within-model diffusion. This method, if corrrected by accept-reject step like in MCMC methods would have been an example of reversible jump.

Certain trandimensional problems can be viewed as marked point processes [112]. In these problems, the variables whose number varies are regarded as marked points. Using the birth-and-death simulation idea of Preston [100] and Ripley [103], Stephens [112] developed a point-process approach for finite mixture analysis. Cappé et al. [23] further extended the point-process idea to compare the reversible jump algorithm with the continuous time birth-and-death samplers.

2.7 Model averaging

As was discussed in Section 2.2, full description of parameter and model uncertainties is necessary for reliable decision-making and quantification of prediction uncertainties. The Bayesian inference methodology provides a natural mechanism for the quantification of all uncertainties. A direct consequence is that future predictions can be based on all models weighted by their posterior probabilities. Making predictions based on all models typically has the effect of lowering bias and improve predictive capability [21]. Further, posterior predictions based on a selected model can be shown to always underestimate prediction uncertainty [27, 102].

Consider that there is a set of M plausible models that are available for the prediction of a quantity of interest Δ. Given the data $\boldsymbol{\mathcal{D}}$, the posterior model probabilities of each model m, $P(m|\boldsymbol{\mathcal{D}})$ is obtained by Bayes's rule. Now, the posterior distribution of the quantity of interest, Δ, is obtained by *Bayesian model averaging* as:

$$p(\Delta|\boldsymbol{\mathcal{D}}) = \sum_{i=1}^{M} p(\Delta|M_i, \boldsymbol{\mathcal{D}})p(M_i|\boldsymbol{\mathcal{D}}). \tag{2.39}$$

If we use a log-loss scoring rule, where the loss function for distribution, $q(\Delta)$, is given by:

$$C(\Delta, q) = -A \log q(\Delta) + B,\ \ A > 0, B > 0, \tag{2.40}$$

the expected loss of approximating the true posterior distribution $p(\Delta|\boldsymbol{\mathcal{D}})$ with $p(\Delta|M_i, \boldsymbol{\mathcal{D}})$ is given by $-\mathbb{E}_{p(\Delta|\boldsymbol{\mathcal{D}})}[\log p(\Delta|M_i, \boldsymbol{\mathcal{D}})]$. Since the Kullback-Leibler divergence between any two distributions is always non-negative, it follows that

$$D_{KL}(p(\Delta|\boldsymbol{\mathcal{D}}) \| p(\Delta|M_i, \boldsymbol{\mathcal{D}})) \geq 0,\ \ i = 1, ..., M \tag{2.41}$$

Consequently,

$$-\mathbb{E}_{p(\Delta|\boldsymbol{\mathcal{D}})}\Big[\log \sum_{i=1}^{M} p(\Delta|M_i, \boldsymbol{\mathcal{D}})p(M_i|\boldsymbol{\mathcal{D}})\Big] \leq -\mathbb{E}_{p(\Delta|\boldsymbol{\mathcal{D}})}\Big[\log p(\Delta|M_i, \boldsymbol{\mathcal{D}})\Big]. \tag{2.42}$$

The expected loss of approximating the uncertainty in Δ based on any single model is always greater than the quantifying the uncertainty by averaging over all plausible models. Thus, a full description of prediction uncertainty is obtained by the posterior-weighted-average of model predictions of the quantity of interest Δ.

Chapter 3

Network inference with adaptive MCMC

We now present the details of a new framework for large-scale inference of chemical reaction networks that transforms the network inference problem into a fixed dimensional sampling problem and uses adaptive Markov chain Monte Carlo for improving sampling efficiency. Viewing network inference as a fixed-dimensional problem allows us to adapt existing fixed-dimensional adaptive MCMC algorithms for network inference. Adaptive MCMC methods adapt proposals based on previous posterior samples and produce improved MCMC simulations without manual tuning. The material of this chapter elaborates on one of our publications [42].

3.1 Reaction networks are nested

As we saw in Chapter 1, the law of mass action gives the rate of a chemical reaction (say $X + Y \to Z$) as the product of a reaction-specific rate constant k with reactant concentrations $[X]$ and $[Y]$.

$$\text{Rate} = -k[X][Y]. \tag{3.1}$$

The rate constant k is expressed in Arrhenius form as

$$k = AT^n \exp\left(-\frac{E_a}{RT}\right), \tag{3.2}$$

where A is the pre-exponential factor, E_a is the activation energy, n is the temperature exponent, R is the universal gas constant, and T is temperature. In this thesis, we treat k as the combined unknown parameter; it is also possible to infer A, E_a, and n separately (given observations over a range of temperatures) but we leave such a treatment for subsequent work.

In any chemically reacting process, the rates of the individual elementary reactions in the reaction network together determine the values of the observables. And reacting flow models are seldom linear; that is, the observables depend nonlinearly on the elementary reaction rates and on the rate constants. Interestingly though, the net species production rate is additive, i.e., the total production/destruction rate of a species is the cumulative sum of the production/destruction rates of the species from all reactions it participates in. From a statistical perspective, this implies that the set of plausible networks given a set of proposed reactions and a noise model, are *nested*. Specifically, a reaction can be eliminated from the network simply by setting the corresponding rate constant to zero. The nested structure of reaction network inference is exploited in two ways in this thesis.

First note that an across-model sampler for nested models can be constructed even with a posterior sampler that operates over a space of *fixed* dimension. For example, consider a setting where we have $N = 5$ postulated elementary reactions with rate constants k_1, k_2, k_3, k_4, and k_5. Thus we wish to compare $2^5 - 1$ networks that are *a priori* plausible. The key idea is to recognize that switching from a network M_i (for instance, comprising reactions 1, 2, and 5) to a network M_j (for instance, comprising reactions 3 and 4) requires that the parameter vector $\boldsymbol{k} \equiv (k_1, k_2, k_3, k_4, k_5)$ change from $(a, b, 0, 0, c)$ to $(0, 0, d, e, 0)$, where a, b, c, d, and e are nonzero rate constants for each reaction. Second, the nested structure of network inference naturally leads to proposing moves between networks such that the rate constants of reactions that are common to the networks retain their parameter values.

The first step in developing a sampling scheme that assigns *zero*-value to rate constants is to impose point-mass mixture priors on the rate constants k_i [73, 88]. For simplicity, in the subsequent numerical demonstrations we will take the priors to be independent in each dimension (i.e., for each reaction), such that $p(\boldsymbol{k}) = \prod_{i=1}^{N} p_i(k_i)$. We note, however, that priors can certainly be designed to reflect any additional information, i.e., knowledge that necessitates the joint inclusion and exclusion of reactions. In any case, a point-mass mixture prior is given by

$$p_i(k_i) = w_{0,i}\delta(k_i) + w_{1,i}\,\mathcal{C}_i(k_i), \tag{3.3}$$

where $w_{0,i}$ and $w_{1,i} = 1 - w_{0,i}$ are weights of the two prior components. $\delta(k_i)$ is a probability atom (a point with unity probability mass) at zero and $\mathcal{C}_i(k_i)$ is the continuous component of the prior distribution. The continuous component of the prior probability distribution describes any prior information about the values that the rate constant can take and is often elicited from experts. If no such information exists, $\mathcal{C}_i(k_i)$ may be a uniform or log-uniform distribution over all positive real numbers (an 'uninformative' prior). In any case, Bayesian inference and indeed our framework allow the model developer significant flexibility in setting the prior distribution based on his or her subjective belief or any pre-existing information. The weights $w_{0,i}$ and $w_{1,i}$ are prior beliefs about reaction i being included or excluded, respectively, from the inferred model. The model developer may use these weights to impose prior information about the importance of this reaction in modeling the reacting flow model output.

It is instructive to discuss two specific cases. First, if the model developer has no prior preference for the inclusion or exclusion of a reaction, then an appropriate choice for the weights is an indifference prior setting of $w_{0,i} = w_{1,i} = 0.5$. In contrast, if the model developer believes that reaction i should definitely be part of the inferred model, then he/she can set $w_{0,i}$ to zero and $w_{1,i}$ to one. Note that if all the reactions are assigned a prior inclusion probability of $w_1 = 1.0$, then the model inference framework reduces to the familiar Bayesian parameter inference problem.

Letting $\boldsymbol{\mathcal{D}}$ denote the available data, an application of Bayes' rule to the parameter vector $\boldsymbol{k}$ yields

$$p(\boldsymbol{k}|\boldsymbol{\mathcal{D}}) \propto p(\boldsymbol{\mathcal{D}}|\boldsymbol{k})p(\boldsymbol{k}). \tag{3.4}$$

Here $p(\boldsymbol{\mathcal{D}}|\boldsymbol{k})$ is viewed as a function of $\boldsymbol{k}$: it is the likelihood function, which reflects the discrepancy between the data $\boldsymbol{\mathcal{D}}$ and the model prediction at the specified $\boldsymbol{k}$. The precise form of the likelihood function depends on the noise model used to describe the data. For instance, in the examples of Section 4.4, we use an additive Gaussian model, yielding

$$\boldsymbol{\mathcal{D}} = \boldsymbol{G}(\boldsymbol{k}) + \boldsymbol{\epsilon}. \tag{3.5}$$

Here $G(\boldsymbol{k})$ is the prediction of the *forward* model (the chemically reacting flow model) at the specified parameter value $\boldsymbol{k}$, and $\boldsymbol{\epsilon}$ reflects a combination of observational noise and model errors. We assume that every component of ϵ_j of ϵ is independent with mean zero and variance σ^2, $\epsilon_j \sim \mathcal{N}(0, \sigma^2)$. Because the prior distribution on each k_i is a point-mass mixture (3.3), the resulting posterior distribution of $\boldsymbol{k}$ is also a mixture distribution over the product space of all reactions, where each component of the mixture contains a different set of non-zero rate constants and thus represents a different model. Sampling the posterior distribution of $\boldsymbol{k}$ implies visiting posterior mixture components in proportion to their probabilities. Therefore, a scheme that samples $p(\boldsymbol{k}|\boldsymbol{\mathcal{D}})$ will not only provide a full description of uncertainties in rate constant values, but will also yield estimates of the posterior model probabilities proportional to the number of times each posterior mixture component is visited.

3.2 Posterior exploration by Markov chain Monte Carlo

The multi-dimensional posterior distribution of the parameter vector $\boldsymbol{k}$ obtained in the last section cannot be sampled directly; because of the nonlinear forward model $\boldsymbol{G}(\boldsymbol{k})$, the likelihood does not have a standard form and certainly is not conjugate to the prior distribution. However, simulating posterior samples is possible using the independence Metropolis-Hastings (MH) algorithm [105, 114]. In an independence Metropolis-Hastings algorithm, the proposal distribution at each step is *independent of the current location of the chain.* Algorithm 3 describes the independence sampler using pseudocode. We note that another commonly-used class of MCMC algorithms, random-walk Metropolis-Hastings, is not suitable for our problem because its chains will tend to remain "stuck" in the point mass component of a parameter posterior unless the support of the continuous component is very close to zero.

Algorithm 3 The independence Metropolis-Hastings algorithm

```
Given: Data $\boldsymbol{\mathcal{D}}$, prior density $p(\boldsymbol{k})$, likelihood function $p(\boldsymbol{\mathcal{D}}|\boldsymbol{k})$, proposal $q(\boldsymbol{k})$, number of steps $T$
Initialize $\boldsymbol{k}^0$
for $t = 0$ to $T - 1$ do
    Sample $u \sim \mathcal{U}_{[0,1]}$
    Sample $\boldsymbol{k}^* \sim q(\boldsymbol{k}^*)$
    if $u < \mathcal{A}(\boldsymbol{k}^t, \boldsymbol{k}^*) = \min\left\{1, \frac{p(\boldsymbol{k}^*|\boldsymbol{\mathcal{D}})q(\boldsymbol{k}^t)}{p(\boldsymbol{k}^t|\boldsymbol{\mathcal{D}})q(\boldsymbol{k}^*)}\right\}$ then
        $\boldsymbol{k}^{t+1} = \boldsymbol{k}^*$
    else
        $\boldsymbol{k}^{t+1} = \boldsymbol{k}^t$
    end if
end for
```

The Metropolis-Hastings algorithm's efficiency in exploring the posterior distribution rests on the design of an effective proposal distribution. "Efficiency" in this context refers to how effectively the Markov chain explores the posterior—i.e., how nearly independent its states are—which translates directly into the Monte Carlo error of a sample-based posterior estimate. A good proposal distribution will require fewer posterior density evaluations to achieve a given error. Recall that computation of the posterior density $p(\boldsymbol{k}|\boldsymbol{\mathcal{D}})$ for a proposed parameter value involves evaluating the likelihood $p(\boldsymbol{\mathcal{D}}|\boldsymbol{k})$, which in turn requires solving the forward model. Restricting the number of forward model solves is especially important in the present application context, because detailed models of chemically reacting flow are computationally expensive.

Since the marginal posterior distribution of each parameter k_i is a mixture of a point mass and continuous components, the proposal distribution for each k_i is taken to be an independent point-mass mixture distribution of the form:

$$q(k_i; \boldsymbol{\psi_i}) = b_{i,0}\delta(k_i) + \sum_{m=1}^{M} b_{i,m} q_m(k_i; \theta_{i,m}). \tag{3.6}$$

In the above equation, $\delta(k_i)$ is a point mass at zero, $q_m(k_i; \theta_{i,m})$ are continuous components of the proposal distribution, and $\bar{\boldsymbol{\psi}} \equiv (b_{i=1:N,m=0:M}, \theta_{i=1:N,m=1:M})$ comprises all the parameters describing the proposal distribution. Recall that N is the number of proposed reactions, and thus the dimension of the posterior distribution. The number of continuous components M in each dimension is a choice left to the user. Increasing M can potentially improve the approximation of the posterior by the proposal, especially if the continuous part of the posterior distribution is itself multimodal. This is desirable, because a good proposal distribution for independence Metropolis-Hastings is generally one that approximates the posterior as closely as possible. But higher values of M increase the number of parameters needed to describe the proposal distribution, which can affect the cost and convergence rate of the proposal adaptation scheme discussed in Section 3.2.1. Choosing an independent proposal distribution for each parameter k_i means that the joint proposal distribution is given by

$$q(\boldsymbol{k}; \bar{\boldsymbol{\psi}}) = \prod_{i=1}^{N} q(k_i; \boldsymbol{\psi_i}). \tag{3.7}$$

It is interesting to note that the above fixed-dimensional Metropolis-Hastings algorithm is in fact a particular choice of the more general reversible jump MCMC described in Section 2.6.4.

3.2.1 Adaptive MCMC by online expectation maximization

As noted above, efficient sampling suggests that we choose the proposal parameters $\bar{\boldsymbol{\psi}}$ so that (3.7) closely approximates the posterior. Of course, the true posterior distribution is not characterized a priori; its exploration is in fact the goal of MCMC. A useful strategy for improving sampling efficiency is, then, to continuously adapt the proposal parameters based on past samples from the MCMC chain. Algorithms of this kind are known as *adaptive MCMC* and require additional theoretical analysis to guarantee convergence to the target distribution [105]. A commonly used adaptation criterion is to tune the proposal parameters to minimize the Kullback-Leibler (KL) divergence from the posterior distribution to the proposal distribution [4, 73]. We adopt this strategy here and detail the adaptive independence Metropolis-Hastings algorithm as follows.

Formally, the optimal proposal parameters are given by

$$\bar{\boldsymbol{\psi}}^* = \arg\min_{\bar{\boldsymbol{\psi}}} \mathcal{D}_{KL}\left(p(\boldsymbol{k}|\boldsymbol{\mathcal{D}})\|q(\boldsymbol{k};\bar{\boldsymbol{\psi}})\right) = \arg\min_{\bar{\boldsymbol{\psi}}} \int p(\boldsymbol{k}|\boldsymbol{\mathcal{D}}) \log\left(\frac{p(\boldsymbol{k}|\boldsymbol{\mathcal{D}})}{q(\boldsymbol{k};\bar{\boldsymbol{\psi}})}\right) d\boldsymbol{k}. \quad (3.8)$$

Since this objective function involves integration over the posterior distribution $p(\boldsymbol{k}|\boldsymbol{\mathcal{D}})$, finding a solution before exploring the posterior is difficult. An effective strategy is to use a stochastic approximation method [104, 79] that couples posterior exploration with the solution of the minimization problem. A generic stochastic approximation method for problem (3.8) involves iteratively *(i)* simulating a batch of samples from the posterior distribution to *estimate* the KL divergence above, then *(ii)* using those results to update the proposal parameters. Under conditions explained by [5], the proposal parameters converge to the optimal solution of (3.8) asymptotically. Within this general procedure, one could consider two possible instantiations. The first is stochastic gradient descent: simulate a finite number of samples from the posterior distribution and use them to compute a noisy estimate of the *gradient* of the objective in (3.8) with respect to $\bar{\boldsymbol{\psi}}$; then take a step in the negative-gradient direction to update the parameters in each iteration. This approach is detailed in the paper by [73]. The second approach involves solving (3.8) using a method called online expectation maximization (EM) [4]. Online EM alternately uses posterior samples to update estimates of the expectation of the logarithm of *complete-data likelihood* (E-step) and then directly adapts the proposal parameters using analytical expressions (M-step). We found the online EM approach to be more robust in practice, and have thus adopted it for this work. (See A.0.3 for more details on the complete-data likelihood.)

Here, we describe the expressions used to update the proposal parameters using the online EM algorithm. A detailed derivation of the online EM algorithm applied to point-mass mixture priors can be found in A. We consider the case where the reaction rate parameter vector is N-dimensional, i.e., $\boldsymbol{k} = (k_1, k_2, \ldots, k_N)$, where T samples are simulated from the posterior distribution $p(\boldsymbol{k}|\boldsymbol{\mathcal{D}})$ between each proposal parameter update, and where the M continuous components of the proposal distribution are Gaussian, resulting in a proposal of the form:

$$q(k_i; \boldsymbol{\psi_i}) = b_{i,0}\delta(k_i) + \sum_{m=1}^{M} b_{i,m}\mathcal{N}_m(k_i; \theta_{i,m}). \tag{3.9}$$

A non-adaptive component $\tilde{q}$ must also be added to the proposal distribution to satisfy conditions for the convergence of the adaptive MCMC algorithm to the posterior distribution (see A.0.3 for details). Thus, the overall proposal in each dimension is given by

$$q_s(k_i) = \lambda_i \tilde{q}(k_i, \tilde{\boldsymbol{\psi}}_{\boldsymbol{i}}) + (1 - \lambda_i) q(k_i; \boldsymbol{\psi_i}), \tag{3.10}$$

where $0 < \lambda_i < 1$ and $\tilde{\boldsymbol{\psi}}_{\boldsymbol{i}}$ is a fixed set of proposal parameter values. At each step n

of the online EM algorithm:

1. Simulate T samples $\boldsymbol{k}^1$, $\boldsymbol{k}^2$, …, $\boldsymbol{k}^T$ from the posterior $p(\boldsymbol{k}|\boldsymbol{\mathcal{D}})$ using an independence Metropolis-Hastings algorithm with the current proposal parameter values.

2. Compute (for all parameters $i = 1 \ldots N$)

   For $m = 0$ to M :

   $O_{i,m} = \dfrac{1}{T}\displaystyle\sum_{t=1}^{T} \gamma(z_{i,m}^t),$

   For $m = 1$ to M :

   $$P_{i,m} = \frac{1}{T}\sum_{\substack{t=1\\k_i^t\neq 0}}^{T} \gamma(z_{i,m}^t), \qquad Q_{i,m} = \frac{1}{T}\sum_{\substack{t=1\\k_i^t\neq 0}}^{T} \gamma(z_{i,m}^t)k_i^t, \qquad R_{i,m} = \frac{1}{T}\sum_{\substack{t=1\\k_i^t\neq 0}}^{T} \gamma(z_{i,m}^t)(k_i^t)^2,$$

   where

   $$\gamma(z_{i,m}^t) = \begin{cases} 1 & \text{if } k_i^t = 0 \text{ and } m = 0 \\ 0 & \text{if } k_i^t = 0 \text{ and } m \neq 0 \\ 0 & \text{if } k_i^t \neq 0 \text{ and } m = 0 \\ \frac{b_{i,m}\mathcal{N}(k_i^t;\mu_{i,m},\sigma_{i,m}^2)}{\sum_{m'=1}^{M} b_{i,m'}\mathcal{N}(k_i^t;\mu_{i,m'},\sigma_{i,m'}^2)} & \text{if } k_i^t \neq 0 \text{ and } m \neq 0. \end{cases} \tag{3.11}$$

3. Set $\eta_n = 1/n$ and update the running posterior summaries as

   $$\begin{aligned} S_n^{O_{i,m}} &= S_{n-1}^{O_{i,m}} + \eta_n(O_m - S_{n-1}^{O_{i,m}}) \\ S_n^{P_{i,m}} &= S_{n-1}^{P_{i,m}} + \eta_n(P_{i,m} - S_{n-1}^{P_{i,m}}) \\ S_n^{Q_{i,m}} &= S_{n-1}^{Q_{i,m}} + \eta_n(Q_{i,m} - S_{n-1}^{Q_{i,m}}) \\ S_n^{R_{i,m}} &= S_{n-1}^{R_{i,m}} + \eta_n(R_{i,m} - S_{n-1}^{R_{i,m}}). \end{aligned} \tag{3.12}$$

4. Solve for new proposal parameters:

$$
\begin{aligned}
b_{i,m} &= \frac{S_n^{O_{i,m}}}{\sum_{m'=0}^{M} S_n^{O_{i,m'}}} \\
\mu_{i,m} &= \frac{S_n^{Q_{i,m}}}{S_n^{P_{i,m}}} \\
\sigma_{i,m}^2 &= \frac{\mu_{i,m}^2 S_n^{P_{i,m}} - 2\mu_{i,m} S_n^{Q_{i,m}} + S_n^{R_{i,m}}}{S_n^{P_{i,m}}}.
\end{aligned}
\tag{3.13}
$$

3.2.2 Random-scan AIMH for nested models

A straightforward application of the adaptive independence MH algorithm described so far has one important inefficiency. In the parameter sampling step of Algorithm 3, all the parameters are proposed jointly and then passed through an accept-reject step. This approach can lead to a very high rate of rejection that consequently renders the adaptation ineffective. The alternative is to use a componentwise independent MH approach, wherein only a single component (or a small block of components) of the parameter vector $\boldsymbol{k}$ is proposed at a time while the other parameters are kept fixed, and the resulting parameter vector is immediately passed through an accept-reject step. We use a random-scan variant of the componentwise MH scheme which randomly selects a block of components to be updated. Proposing to update only a few parameter values while keeping the other parameter values fixed is a natural choice for nested models and generally produces good exploration of the posterior distribution. Updating only one or a few components at a time also provides the practical benefit of a local search, since this amounts to making small jumps in the model space. Algorithm 4 summarizes the overall algorithm we use to generate samples from the posterior distribution. In implementing this algorithm for the following examples, we choose $T = 1000$ and $l = 1$ or 2.

Algorithm 4 Model inference by the adaptive independence Metropolis-Hastings algorithm

```
Given: Data $\boldsymbol{\mathcal{D}}$, prior density $p(\boldsymbol{k})$, likelihood function $p(\boldsymbol{\mathcal{D}}|\boldsymbol{k})$, proposal distributions $q_s(k_i) = \lambda_i \tilde{q}(k_i; \tilde{\boldsymbol{\psi_i}}) + (1 - \lambda_i) q(k_i; \boldsymbol{\psi_i})$, number of proposal updates $N_{iter}$, number of samples $T$ between proposal updates
Initialize starting point $\boldsymbol{k}^0$ and proposal parameters $\bar{\boldsymbol{\psi}}_{\mathbf{0}}$
for $n = 1$ to $N_{iter}$ do
    for $t = 1$ to  $T$ do
        Select the number of parameters $l \ll N$ to be updated.
        Randomly select $l$ parameter indices: $r_1$, $r_2$,. . ., $r_l < N$
        Sample $u \sim \mathcal{U}_{[0,1]}$
        for $p = 1$ to $l$ do
            Sample $k^*_{r_p} \sim q_s(k^*_{r_p})$
        end for
        Set $k^*_{r_p} = k^{t-1}_{r_p}$ for $r_p \setminus \{r_1, r_2, \dots, r_l\}$
        if $u < \mathcal{A}(\boldsymbol{k}^{t-1}, \boldsymbol{k}^*) = \min\{1, \frac{p(\boldsymbol{k}^*|\boldsymbol{\mathcal{D}}) q_s(\boldsymbol{k}^{t-1})}{p(\boldsymbol{k}^{t-1}|\boldsymbol{\mathcal{D}}) q_s(\boldsymbol{k}^*)}\}$ then
            $\boldsymbol{k}^t = \boldsymbol{k}^*$
        else
            $\boldsymbol{k}^t = \boldsymbol{k}^{t-1}$
        end if
    end for
    Update summary statistics $\boldsymbol{S_n^{O_{1:N,0:M}}}$, $\boldsymbol{S_n^{P_{1:N,1:M}}}$, $\boldsymbol{S_n^{Q_{1:N,1:M}}}$, and $\boldsymbol{S_n^{R_{1:N,1:M}}}$
    Update proposal parameters $\bar{\boldsymbol{\psi}}_{\boldsymbol{n}}$: $\boldsymbol{b_{1:N,0:M}}$, $\boldsymbol{\mu_{1:N,1:M}}$, and $\boldsymbol{\sigma^2_{1:N,1:M}}$
    Store $\boldsymbol{k}^{1:T}$ and reset $\boldsymbol{k}^0 \leftarrow \boldsymbol{k}^T$
end for
```

3.3 Numerical demonstrations: catalytic reforming of methane

We demonstrate the approach formulated in the preceding sections on three example problems. In particular, we infer chemical kinetic models for steam and dry reforming of methane catalyzed by rhodium. The first problem uses synthetic data to demonstrate the consistency of the Bayesian model inference procedure, while the second and third examples use experimental data drawn from the literature. Methane reforming is an important process because it provides an effective route for the industrial production of syngas (CO+H_2). Catalytic reforming of methane has been studied previously, and a few kinetic models have been proposed [33, 86, 85, 63]. The development of these models has proceeded by collecting possible elementary reactions and making educated guesses about the appropriate pathways, with the selection of rate parameter values based on existing literature or fits to experimental data.

One of the most common experimental configurations for studying catalytic reactions is a stagnation flow reactor. Stagnation flow reactors provide favorable fluid-mechanical properties that enable measurement of the gas-phase boundary layer near the catalytic surface. Hence we use gas-phase measurements from stagnation flow reactors as data for our inference procedure. Recall that experimental data enters the Bayesian inference formulation through the likelihood function $p(\boldsymbol{\mathcal{D}}|\boldsymbol{k})$ (Section 3). The likelihood function must also therefore incorporate a detailed numerical model of the stagnation flow reactor in order to compare the data with predictions based on any candidate kinetic model. We begin by discussing this reactor model.

3.3.1 Stagnation flow reactor model

The boundary layer flow equations in a stagnation flow reactor (schematic in Figure 3-1) can be modeled as a one-dimensional axisymmetric flow using similarity reduction [77]. The stagnation-flow reactor boundary layer equations have been used by a number of authors in studies of catalytic surface reactions [33, 86, 85]. The governing equations are:

$$
\begin{aligned}
\frac{d(\rho u)}{dz} + 2\rho V &= 0 && (3.14)\\
\rho u \frac{dV}{dz} + \rho V^2 &= -\Lambda_r + \frac{d}{dz}\left(\mu \frac{dV}{dz}\right) && (3.15)\\
\rho u c_p \frac{dT}{dz} &= \frac{d}{dz}\left(\lambda \frac{dT}{dz}\right) - \sum_{\alpha=1}^{K_g} \rho Y_\alpha V_\alpha c_{p\alpha} \frac{dT}{dz} - \sum_{\alpha=1}^{K_g} h_\alpha W_\alpha \dot{\omega}_\alpha && (3.16)\\
\rho u \frac{dY_\alpha}{dz} &= -\frac{d}{dz}(\rho Y_\alpha V_\alpha) + W_\alpha \dot{\omega}_\alpha, \quad (\alpha = 1 \ldots K_g) && (3.17)\\
\dot{s}_\beta &= 0, \quad (\beta = 1 \ldots K_s) && (3.18)
\end{aligned}
$$

$$p = \rho RT \sum_{\alpha=1}^{K_g} \frac{Y_\alpha}{W_\alpha} \tag{3.19}$$

In the above equations, the axial spatial coordinate z is the independent variable, while the axial velocity u, the scaled radial velocity V, the fluid temperature T, and the species mass fractions Y_α are the dependent variables. The pressure-gradient eigenvalue is

$$\Lambda_r = \frac{1}{r}\frac{dp}{dr}. \tag{3.20}$$

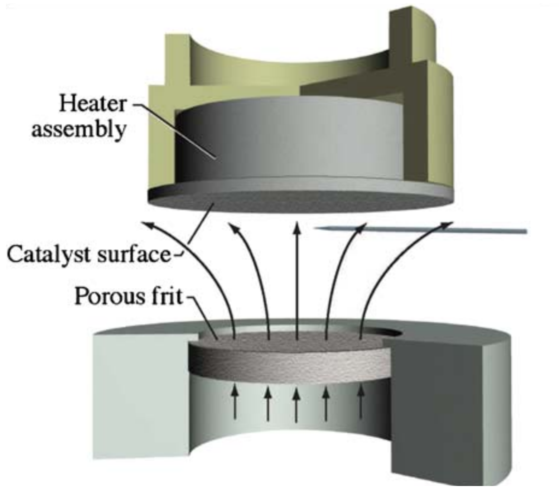


Figure 3-1: Stagnation flow reactor; figure reproduced from [86].

The perfect gas equation (3.19) relates the pressure p to the temperature T, density ρ, and the species mass fractions at any point. In equations (3.14)–(3.18), μ is the fluid dynamic viscosity, λ is the thermal conductivity, c_p is the mixture specific heat, $c_{p\alpha}$ are species specific heats, h_α are species specific enthalpies, and W_α are the molecular weights of the species. $\dot{\omega}_\alpha$ denotes the molar production rate of the gas-phase species indexed by α, and $\dot{s}_\beta$ the production rate of the surface species, indexed by β. There are K_g gas-phase species and K_s surface species. A detailed chemical kinetic model is used to compute the species production rates $\dot{\omega}_\alpha$ and $\dot{s}_\beta$.

We assume that every candidate detailed chemical kinetic model involving N reactions among these species can be represented in the general form

$$\sum_{j=1}^{K_g+K_s} \nu'_{j,i} X_j \longleftrightarrow \sum_{j=1}^{K_g+K_s} \nu''_{j,i} X_j, \quad (i = 1 \ldots N), \tag{3.21}$$

where $\nu_{j,i}$ are integer stoichiometric coefficients and X_j is the chemical name of the jth species. The molar production rates $\dot{\omega}_\alpha$ and $\dot{s}_\beta$ are summations over all reactions:

$$\dot{\omega}_\alpha = \sum_{i=1}^{N} \nu_{\alpha,i} q_i, \quad \dot{s}_\beta = \sum_{i=1}^{N} \nu_{\beta,i} q_i, \tag{3.22}$$

where

$$\nu_{\alpha,i} = \nu''_{\alpha,i} - \nu'_{\alpha,i}, \tag{3.23}$$

and similarly for $\nu_{\beta,i}$. The rate of progress q_i of the ith reaction, which is assumed to obey mass-action kinetics, is the difference between the forward and reverse reaction rates:

$$q_i = k_{i,f} \prod_{j=1}^{K_g+K_s} [X_j]^{\nu'_{j,i}} - k_{i,b} \prod_{j=1}^{K_g+K_s} [X_j]^{\nu''_{j,i}}. \tag{3.24}$$

The form of the concentrations $[X_j]$ in (3.24) depends on whether the species is in gas phase or on the surface. Also, it is known from earlier work [85] that species production rates due to purely gas-phase reactions are negligible at normal operating conditions. Thus we omit purely gas-phase reactions when evaluating $\dot{\omega}_\alpha$ in our differential equation model.

The species diffusion velocities are computed using a multicomponent diffusion model as

$$V_\alpha = \frac{1}{X_\alpha \bar{W}} \sum_{j \neq \alpha}^{K_g} W_j D_{\alpha,j} \frac{dX_j}{dz} - \frac{D_\alpha^T}{\rho Y_\alpha} \frac{1}{T} \frac{dT}{dz}. \tag{3.25}$$

Here X_α and X_j are the species mole fractions, $\bar{W}$ is the mean molecular weight, $D_{\alpha,j}$ are multicomponent diffusion coefficients, and D_α^T are thermal diffusion coefficients. At the reactor inlet, boundary conditions are

$$u = U_{in}, \; V = 0, \; T = T_{in}, \; Y_\alpha = Y_{\alpha,in}, \tag{3.26}$$

and at the catalytic stagnation surface, the boundary conditions are

$$u = 0,\ V = 0,\ T = T_s,\ \rho Y_\alpha V_\alpha = F_{\mathrm{cg}} \dot{\omega}_\alpha W_\alpha. \tag{3.27}$$

The boundary condition in (3.27) states that the gas-phase species diffusion flux at the stagnation surface is balanced by species consumption by catalytic reactions. The boundary condition also contains a parameter F_{cg}, which specifies the effective catalyst area. Since the catalyst particles are dispersed in a porous medium, the effective catalyst area A_{catalyst} is much greater than the geometric area $A_{\mathrm{geometric}}$ of the stagnation surface. The parameter F_{cg} is defined as

$$F_{\mathrm{cg}} = \frac{A_{\mathrm{catalyst}}}{A_{\mathrm{geometric}}} \tag{3.28}$$

The steady-state stagnation flow axisymmetric boundary layer equations form a system of ordinary differential equations. These equations are discretized using a finite difference method and the resulting algebraic equations are solved using a combination of pseudo-time marching and Newton's method [77]. We use the chemically reacting flow software package Cantera 2.0.2 [54] to compute species production rates and to solve the steady-state stagnation flow axisymmetric boundary layer equations.

3.3.2 Proposed elementary reactions

Beginning with the work of Hickman et al. [68], kinetic models for reactions of methane on rhodium have been developed via a combination of theoretical methods, fits to available experimental data, and previous analyses of related species. [33, 85, 63]. Activation energies for surface reactions are often estimated using the semi-empirical unity bond index-quadratic exponential potential (UBI-QEP) method. The determination of pre-exponential factors, however, has largely relied on fits to observed data or the assignment of nominal values. The uncertainty associated with these rate determination techniques and limited understanding of the associated catalytic reaction pathways make this system a good candidate for Bayesian model inference.

The set of proposed elementary reactions we use in our inference demonstrations is taken from a comprehensive model proposed by McGuire et al. [85] recently and is shown in Table 3.1. The reaction set consists of 42 irreversible elementary reactions involving 12 surface-adsorbed species and gas-phase species. We retain the rate parameters given by McGuire et al. [85] as base values for all reactions, except the following two:

$$\mathrm{CO^* + H^* \rightarrow HCO^* + {}^*}$$

$$\mathrm{HCO^* + {}^* \rightarrow CH^* + O^*}$$

The pre-exponential factor of the first reaction above is changed from 5.0×10^{19} to 5.0×10^{18}, while that of the second reaction is changed from 3.7×10^{24} to 3.7×10^{23}. These changes yield minor improvements in agreement with data at the nominal parameter values. The pre-exponential factors were previously assigned nominal values. The activation energies were estimated by UBI-QEP method, which has an expected error of 1–3 kcal/mol [63].

The surface reactions shown in Table 3.1 are of two different types: adsorption/desorption of gas-phase species and reactions among surface intermediates. In this work, we do not consider the adsorption/desorption reactions (Reactions 31–42) to be uncertain; rather, they are included in all models inferred. In the table, the adsorption/desorption reactions are shaded pink, while the surface reactions we consider to be uncertain are shaded green. In the three examples to be presented below, we treat the thermodynamic properties of the surface species as fixed (i.e., not uncertain). Although the thermodynamic properties are not precisely known, they are fixed indirectly through the individual forward and reverse rate constants. The base values of the forward and reverse rate constants, k_f and k_b, were originally established to satisfy approximate thermodynamic reversibility. Therefore, with the thermodynamic properties fixed, we need only to specify the prior distribution and apply the model inference framework on the forward reactions. The reverse rate constant of each reaction is then

$$k_b = \frac{k_f}{K_{eq}} = \frac{k_b^*}{k_f^*} k_f, \tag{3.29}$$

where K_{eq}, the equilibrium constant of the reaction, is a function of the thermodynamic properties of the species participating in the reaction. In the above equations, k_f and k_b are the perturbed rate constants of the reactions while k_f^* and k_b^* are the base rate constants.

3.3.3 Setup of the Bayesian model inference problem

Before discussing the three model inference examples individually, we describe the choices we make for the likelihood function and prior distribution in our Bayesian formulation. In the following, we use $\tilde{k}$ to refer to the rate constants of the reactions that are treated as uncertain and $\hat{k}$ to denote the rate constants of reactions that are kept fixed. By "fixed," we mean that a particular reaction is always included in the model and that its rate constant is not a target of the inference procedure.

Likelihood function

As described in Section 3.1, evaluating the posterior probability in the Bayesian approach requires evaluating the likelihood function $p(\boldsymbol{\mathcal{D}}|\boldsymbol{k})$, where $\boldsymbol{\mathcal{D}}$ are the data and $\boldsymbol{k} = (\tilde{k}, \hat{k})$ are the reaction parameters. We employ an i.i.d. additive Gaussian model for the difference between model predictions and observations; thus the data are represented as

$$\boldsymbol{\mathcal{D}} = \boldsymbol{G}(\tilde{k}, \hat{k}) + \boldsymbol{\epsilon_n}, \tag{3.30}$$

where $\boldsymbol{\epsilon_n} \sim \mathcal{N}_n(\boldsymbol{0}, \sigma^2 \boldsymbol{I_n})$, n is the number of observations, $\boldsymbol{I_n}$ is an n-by-n identity matrix, and $\boldsymbol{G}(\tilde{k}, \hat{k})$ is the prediction of the forward model at the given value of the reaction parameters. We let the noise standard deviation σ be 0.005. The deterministic predictions $\boldsymbol{G}(\tilde{k}, \hat{k})$ are obtained with the stagnation flow reactor model explained

	Reaction	A	E_a	Uncertainty applied[†]
1	$H^* + O^* \rightarrow OH^* + *$	5.0×10^{22}	83.7	$\log_{10} k = \log_{10} k^* + \mathcal{U}[-2, 2]$
2	$H^* + OH^* \rightarrow H_2O^* + *$	3.0×10^{20}	33.5	$\log_{10} k = \log_{10} k^* + \mathcal{U}[-2, 2]$
3	$OH^* + OH^* \rightarrow H_2O^* + O^*$	3.0×10^{21}	100.8	$\log_{10} k = \log_{10} k^* + \mathcal{U}[-2, 2]$
4	$CO^* + O^* \rightarrow CO_2^* + *$	5.5×10^{18}	121.6	$\log_{10} k = \log_{10} k^* + \mathcal{U}[-2, 2]$
5	$CH_4^* + * \rightarrow CH_3^* + H^*$	3.7×10^{21}	61.0	$\log_{10} k = \log_{10} k^* + \mathcal{U}[-2, 2]$
6	$CH_3^* + * \rightarrow CH_2^* + H^*$	3.7×10^{24}	103.0	$\log_{10} k = \log_{10} k^* + \mathcal{U}[-2, 2]$
7	$CH_2^* + * \rightarrow CH^* + H^*$	3.7×10^{24}	100.0	$\log_{10} k = \log_{10} k^* + \mathcal{U}[-2, 2]$
8	$CH_4^* + O^* \rightarrow CH_3^* + OH^*$	1.7×10^{24}	80.34	$\log_{10} k = \log_{10} k^* + \mathcal{U}[-2, 2]$
9	$CH_3^* + O^* \rightarrow CH_2^* + OH^*$	3.7×10^{24}	120.31	$\log_{10} k = \log_{10} k^* + \mathcal{U}[-2, 2]$
10	$CH_2^* + O^* \rightarrow CH^* + OH^*$	3.7×10^{24}	114.5	$\log_{10} k = \log_{10} k^* + \mathcal{U}[-2, 2]$
11	$CH^* + * \rightarrow C^* + H^*$	3.7×10^{21}	21.0	$\log_{10} k = \log_{10} k^* + \mathcal{U}[-2, 2]$
12	$CH^* + O^* \rightarrow C^* + OH^*$	3.7×10^{21}	30.13	$\log_{10} k = \log_{10} k^* + \mathcal{U}[-2, 2]$
13	$C^* + O^* \rightarrow CO^* + *$	5.2×10^{23}	97.9	$\log_{10} k = \log_{10} k^* + \mathcal{U}[-2, 2]$
14	$CO^* + H^* \rightarrow HCO^* + *$	5.0×10^{18}	108.9	$\log_{10} k = \log_{10} k^* + \mathcal{U}[-2, 2]$
15	$HCO^* + * \rightarrow CH^* + O^*$	3.7×10^{23}	59.5	$\log_{10} k = \log_{10} k^* + \mathcal{U}[-2, 2]$
16	$OH^* + * \rightarrow H^* + O^*$	3.0×10^{20}	37.7	$k = k^*_{16} k_1 / k^*_1$
17	$H_2O^* + * \rightarrow H^* + OH^*$	5.0×10^{22}	106.4	$k = k^*_{17} k_2 / k^*_2$
18	$H_2O^* + O^* \rightarrow OH^* + OH^*$	3.0×10^{21}	171.8	$k = k^*_{18} k_3 / k^*_3$
19	$CO_2^* + * \rightarrow CO^* + O^*$	3.0×10^{21}	115.3	$k = k^*_{19} k_4 / k^*_4$
20	$CH_3^* + H^* \rightarrow CH_4^* + *$	3.7×10^{21}	51.0	$k = k^*_{20} k_5 / k^*_5$
21	$CH_2^* + H^* \rightarrow CH_3^* + *$	3.7×10^{23}	44.1	$k = k^*_{21} k_6 / k^*_6$
22	$CH^* + H^* \rightarrow CH_2^* + *$	3.7×10^{21}	68.0	$k = k^*_{22} k_7 / k^*_7$
23	$CH_3^* + OH^* \rightarrow CH_4^* + O^*$	3.7×10^{21}	24.27	$k = k^*_{23} k_8 / k^*_8$
24	$CH_2^* + OH^* \rightarrow CH_3^* + O^*$	3.7×10^{21}	15.06	$k = k^*_{24} k_9 / k^*_9$
25	$CH^* + OH^* \rightarrow CH_2^* + O^*$	3.7×10^{21}	36.82	$k = k^*_{25} k_{10} / k^*_{10}$
26	$C^* + H^* \rightarrow CH^* + *$	3.7×10^{21}	172.8	$k = k^*_{26} k_{11} / k^*_{11}$
27	$C^* + OH^* \rightarrow CH^* + O^*$	3.7×10^{21}	136.0	$k = k^*_{27} k_{12} / k^*_{12}$
28	$CO^* + * \rightarrow C^* + O^*$	2.5×10^{21}	169.0	$k = k^*_{28} k_{13} / k^*_{13}$
29	$HCO^* + * \rightarrow CO^* + H^*$	3.7×10^{21}	0.0	$k = k^*_{29} k_{14} / k^*_{14}$
	θ^*_{CO}		50.0[b]	
30	$CH^* + O^* \rightarrow HCO^* + *$	3.7×10^{21}	167.5	$k = k^*_{30} k_{15} / k^*_{15}$
31	$H_2 + * + * \rightarrow H^* + H^*$	1.0×10^{-2a}	0.0	-
32	$O_2 + * + * \rightarrow O^* + O^*$	1.0×10^{-2a}	0.0	-
33	$CH_4 + * \rightarrow CH_4^*$	8.0×10^{-3a}	0.0	-
34	$H_2O + * \rightarrow H_2O^*$	1.0×10^{-1a}	0.0	-
35	$CO_2 + * \rightarrow CO_2^*$	4.8×10^{-2a}	0.0	-
36	$CO + * \rightarrow CO^*$	5.0×10^{-1a}	0.0	-
37	$H^* + H^* \rightarrow H_2 + * + *$	3.0×10^{21}	77.8	-
38	$O^* + O^* \rightarrow O_2 + * + *$	1.3×10^{22}	355.2	-
	θ^*_O		-280.0[b]	
39	$CH_4^* \rightarrow CH_4 + *$	1.9×10^{14}	25.1	-
40	$H_2O^* \rightarrow H_2O + *$	3.0×10^{13}	45.0	-
41	$CO_2^* \rightarrow CO_2 + *$	4.1×10^{11}	18.0	-
42	$CO^* \rightarrow CO + *$	3.5×10^{13}	133.4	-
	θ^*_{CO}		-15.0[b]	-

[†] Arrhenius rate expression for k^*(base value): $k^* = A \exp(-E_a/RT)$

[a] Sticking coefficient

[b] Coverage-dependent activation energy

[b] Forward-backward reaction pair I consists of reactions I and I+15

Table 3.1: Proposed reactions for reforming of methane

in Section 3.3.1. The likelihood function is thus given by

$$\begin{aligned} p(\boldsymbol{\mathcal{D}}|\boldsymbol{k}) &= \mathcal{N}_n(\boldsymbol{\mathcal{D}}|\boldsymbol{G}(\tilde{k},\hat{k}),\sigma^2\boldsymbol{I_n}) \\ &= \prod_{t=1}^{n}\mathcal{N}(\boldsymbol{\mathcal{D}}|\boldsymbol{G}(\tilde{k},\hat{k}),\sigma^2) \\ &= \prod_{t=1}^{n}\frac{1}{\sqrt{2\pi\sigma^2}}\exp\left(-\frac{(d^t-\boldsymbol{G}(\tilde{k},\hat{k}))^2}{2\sigma^2}\right), \end{aligned} \tag{3.31}$$

where d^t are components of the data vector $\boldsymbol{\mathcal{D}}$.

Prior specification

The prior distribution in Bayesian analysis should encapsulate information about models and parameters that is available before assimilation of the data presently at hand. Often the priors come from known scientific principles and physical constraints on the parameters. In the context of chemical kinetics, the continuous component of the prior distribution may also derived from previous investigations of the reactions. Furthermore, as described in Section 3.1, priors may also be shaped by expert elicitation [45] or chosen to reflect relative ignorance about the rate parameter values. In our demonstrations we will choose relatively uninformative priors by allowing the rate constants $\boldsymbol{k}$ to vary by two orders of magnitude above and below their base values. Other prior choices, e.g., an exponential distribution or a uniform distribution between zero and some positive upper bound, would also be reasonable. In the same way, prior information about model structure—applied in the form of prior weights on reaction inclusion or exclusion—can also be designed to reflect an investigator's belief about the role or importance of particular reactions in the chemical process.

To illustrate the impact of the prior, we consider three different prior specifications in our numerical demonstrations:

- Prior 1: $p(k_{i,f}) = 0.2\delta(k_{i,f}) + 0.8\mathcal{C}(k_{i,f})$,
- Prior 2: $p(k_{i,f}) = 0.5\delta(k_{i,f}) + 0.5\mathcal{C}(k_{i,f})$,

- Prior 3: $p(k_{i,f}) = 0.8\delta(k_{i,f}) + 0.2\mathcal{C}(k_{i,f})$,

The prior distributions above are imposed identically on *each* reaction. Since reaction rate constants must be positive, while their uncertainties may multiple orders of magnitude, we take the continuous component of each prior distribution to be a bounded uniform distribution on the logarithm of the rate constant. Specifically, we set each $\mathcal{C}(k_{i,f})$ to

$$\mathcal{C}(k_{i,f}) : \log_{10} k_{i,f} \sim \mathcal{U}(\log_{10} k^*_{i,f} - 2, \log_{10} k^*_{i,f} + 2), \tag{3.32}$$

where each $k^*_{i,f}$ above is the base value of the ith forward rate constant. For simplicity, the priors used here are all of same family and width, but in general, one could certainly endow each of the 15 rate constants with distinct priors. One could even encode prior correlations among the rate constants.

The three prior specifications above reflect different prior beliefs in the size and sparsity of the reaction mechanism. Prior 1, with a weight of 0.8 on the continuous component, has a tendency to favor kinetic models with more reactions. Prior 2 is the *indifference* prior with no preference for inclusion or exclusion of reactions; it is equivalent to a uniform prior distribution on the space of all 2^N possible models, and thus allows the data to completely determine the most probable set of reactions. Prior 3 favors smaller models, and is an example of a *sparsity-promoting* prior. Such priors introduce additional parsimony in model structure, over and above the penalty on model complexity automatically imposed by the Bayesian Occam's razor. By using priors that favor reaction exclusion, the posterior distribution over models is biased towards simpler reaction network structures; this has the potential of improving prediction accuracy over unobserved data [64].

3.3.4 Example 1: Steam reforming of methane with synthetic data

In this first example, we infer kinetic models for steam reforming of methane from data generated using a known model. The goal of this example is to demonstrate the consistency of the Bayesian model inference process and to examine the impact of varying amounts of data. We create four synthetic (nested) data sets increasing in size from 10, 20, 40, to 60 points. The data are mole fractions of gas-phase species (H_2, H_2O, CH_4, CO, and CO_2) measured at different locations inside the stagnation flow reactor, at up to three different catalyst surface temperatures T_s. Data set 1 consists of mole fractions 0.1 mm and 0.7 mm from the catalyst surface, while data sets 2, 3, and 4 contain measurements performed 0.1 mm, 0.7 mm, 1.3 mm, and 2 mm from the catalyst surface. Further details on each data set are given in Table 3.2. We generate the data using a kinetic model that contains all the reactions shown in Table 3.1, except reaction pairs (4)–(19) and (6)–(21). Samples of Gaussian noise with mean zero and standard deviation $\sigma = 0.005$ are added to these model predictions to simulate noisy experimental observations. For the purpose of this example, we allow only four reaction pairs to have uncertain parameters and to be candidates for inclusion/exclusion. The other reactions are kept fixed at their base values for the likelihood calculation. The uncertain reaction pairs are shown in Table 3.3.

Data set	Number of data points	Catalyst temperatures
1	10	740°C
2	20	740°C
3	40	740°C, 790°C
4	60	740°C, 790°C, 840°C

Table 3.2: Synthetic data sets for Example 1.

Because we consider only four reaction pairs to have inclusion/exclusion uncertainty, the number of possible models in the present example is $2^4 = 16$. We employ the indifference prior (Prior 2) described in the previous section. 200 000 samples are

Reaction pair†		Reaction
1	(1)–(16)	$H^* + O^* \leftrightarrow OH^* + {}^*$
2	(4)–(19)	$CO^* + O^* \leftrightarrow CO_2{}^* + {}^*$
3	(5)–(20)	$CH_4{}^* + {}^* \leftrightarrow CH_3{}^* + H^*$
4	(6)–(21)	$CH_3{}^* + {}^* \leftrightarrow CH_2{}^* + H^*$

†Reaction pair numbering in the leftmost column is specific to Example 1.

Table 3.3: Proposed reactions for inference in Example 1.

then simulated from the posterior distribution of $\tilde{k}$, using adaptive MCMC, for each of the four data sets. We begin adaptation after generating the first 20 000 samples and discard the next 20 000 samples as burn-in, while the proposal parameters adapt. The most probable models and their probabilities are computed using the remaining 160 000 samples; these are shown in Table 3.4.

Data set 1		Data set 2	
Reaction pairs included	Probability	Reaction pairs included	Probability
1, 3	0.281	1, 3	0.375
1, 4	0.256	1, 3, 4	0.197
1, 3, 4	0.165	1, 4	0.195
1	0.146	1	0.082
1, 2, 3	0.056	1, 2, 3	0.073

Data set 3		Data set 4	
Reaction pairs included	Probability	Reaction pairs included	Probability
1, 3	0.482	1, 3	0.525
1, 2, 3	0.316	1, 2, 3	0.253
1, 3, 4	0.122	1, 3, 4	0.152
1, 2, 3, 4	0.072	1, 2, 3, 4	0.070
1, 4	0.006	1, 4	0.001

Table 3.4: The five most probable models and their probabilities, from Example 1.

We see from the inference results that the data-generating model (i.e., the "true" model) is selected with highest posterior probability for every data set. Although it is possible in principle for the true model not to be assigned the highest posterior probability for finite data [16], we notice here that the true model is always preferred and moreover that its probability increases with more data. This trend also demonstrates the diminishing impact of the prior distribution as more data are included. Indeed, Bayesian model inference is known to be asymptotically consistent, i.e., the posterior concentrates on the true model given infinite data [12], provided that true model is within the set of models being considered.

3.3.5 Example 2: Steam reforming of methane with real data

The second example considers inference of chemical kinetic models for steam reforming of methane using real experimental data from a stagnation flow reactor apparatus [86]. The operating conditions of the experiment are given in Table 3.5; further specifics on the experimental data set (e.g., species and measurement locations) can be found in [86].

In this example, we consider all three prior specifications (Section 3.3.3) imposed on all 15 of the uncertain reaction pairs in Table 3.1. Using the adaptive MCMC procedure of Section 3.2.1, we generate 200 000 samples from the posterior distribution of $\tilde{k}$. Again, we begin adaptation after generating the first 20 000 samples and discard the next 20 000 samples as burn-in. Posterior model probabilities are estimated from the remaining 160 000 samples.

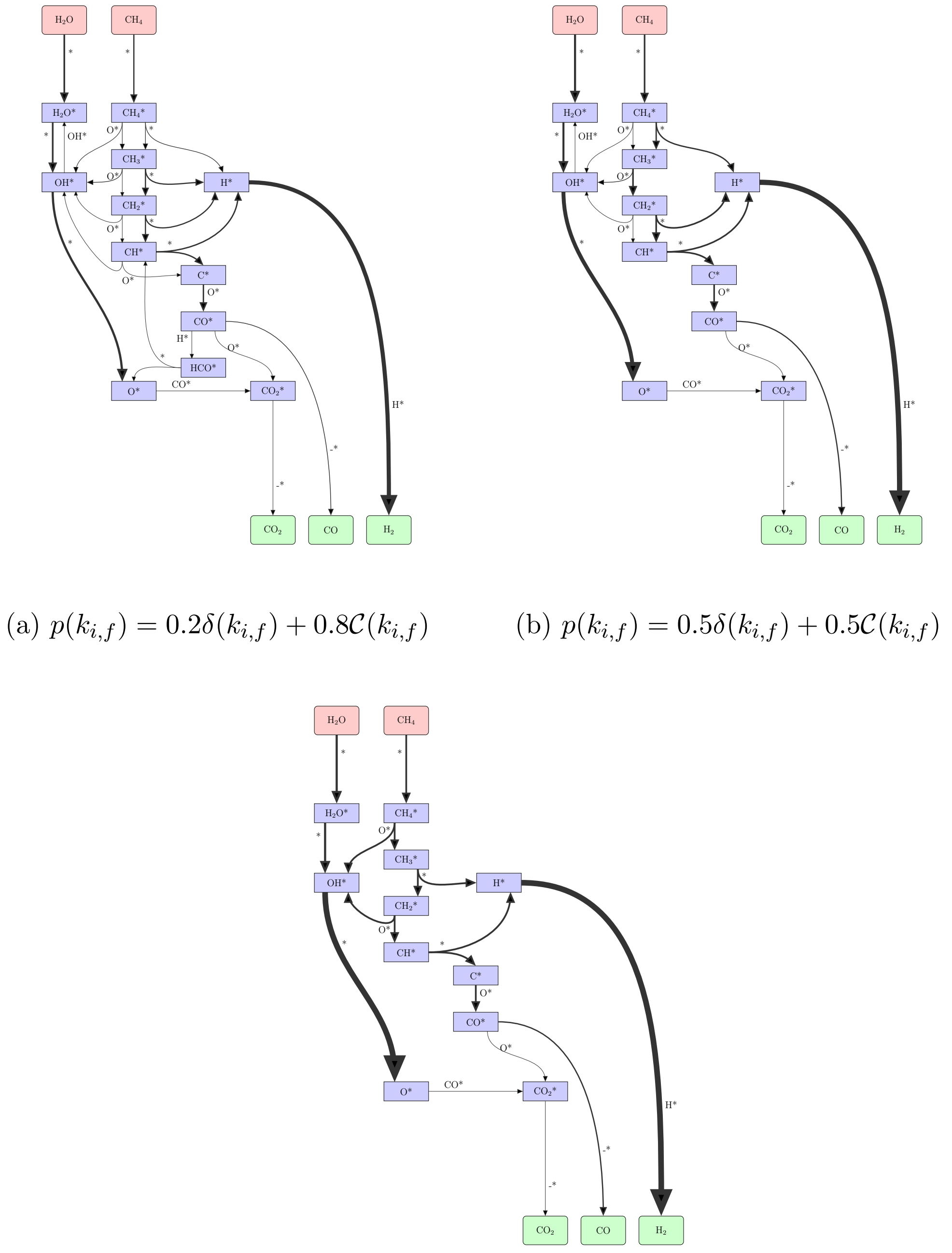
Table 3.6 shows the ten most probable models for each prior specification, and their corresponding frequency (in a total population of 160 000 samples). As expected, the sparsity of the most probable models increases with the weight on the δ-component in the prior. In the case of Prior 1, the model that includes all the reactions is strongly preferred. For Prior 2, the most probable model includes all the reactions except pairs 6–21, 12–27, and 14–29. The exclusion of these three reaction pairs, particularly in the case of an indifference prior, is an example of the Bayesian Occam's razor in action. Prior 3 results in extremely sparse models. Reaction networks corresponding to the highest probability models for the three prior settings are shown in Figure 3-2.

Condition	Value
Inlet composition (by mole fractions)	4.3% CH_4 and 5.9% H_2O (balance Ar)
Inlet temperature	135°C
Catalyst surface temperature	740°C
Inlet velocity	1.3 m/s
Reactor pressure	300 Torr
F_{cg}	20

Table 3.5: Experimental operating conditions for Example 2, from [86].

We also show in Figure 3-3 the posterior marginal inclusion probability of each reaction pair. Since the marginal inclusion probability of a reaction is the average of its inclusion indicator over all possible models in the posterior, it provides a measure of how strongly an individual reaction is supported by the available data. In all three panels of Figure 3-3, we note that the posterior inclusion probability of reaction pair 3–18 is identical to its prior inclusion probability. Reaction pairs 14–29 and 15–30 (reactions involving species HCO*) also have a negligible difference between their prior and posterior inclusion probabilities. These results suggest that the data are not informative about these reactions; in other words, these reactions seem to have an insignificant effect on the level of agreement between model predictions and the available data. Invoking a further principle of parsimony, it may thus be prudent to exclude reaction pairs 3–18, 14–29, and 15–30 from the predictive model, or to reassess their importance by collecting more data.

Figure 3-3 also shows that the posterior marginal inclusion probabilities of reaction pairs 1–16, 2–17, 4–19, and 13–28 remain close to one for each prior specification; these reactions are thus the most strongly supported by available data. The inclusion of reaction pair 4–19 with probability one in all the inferred models is confirmation that



(a) $p(k_{i,f}) = 0.2\delta(k_{i,f}) + 0.8\mathcal{C}(k_{i,f})$

(b) $p(k_{i,f}) = 0.5\delta(k_{i,f}) + 0.5\mathcal{C}(k_{i,f})$

(c) $p(k_{i,f}) = 0.8\delta(k_{i,f}) + 0.2\mathcal{C}(k_{i,f})$

Figure 3-2: Reaction networks of the highest posterior probability models for steam reforming of methane (Example 2), under different prior specifications. Edge thicknesses are proportional to reaction rates calculated using posterior mean values of the rate parameters.

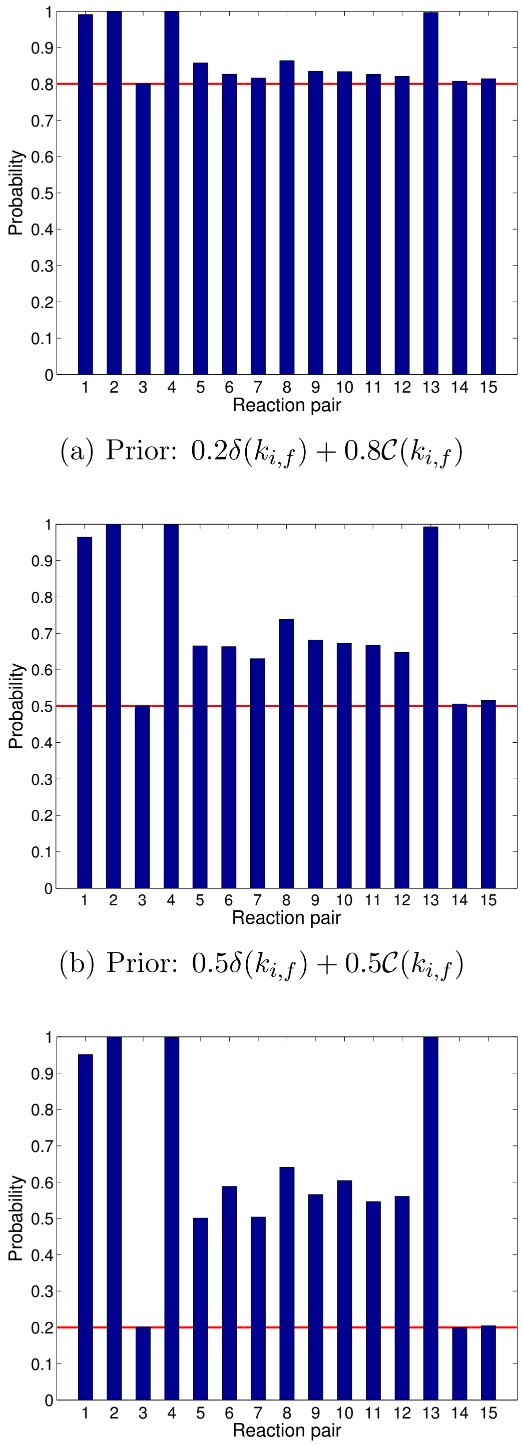
the inference procedure is working well, in that it does not exclude reactions that are absolutely necessary for production of the observed product species. Among all the uncertain reaction pairs, reactions 4 and 19 are the only pair containing CO_2^*, and their inclusion ensures that the model produces CO_2 as one of the products. In general, the near-unity posterior probabilities suggest that all four of these reaction pairs are critical to explaining the steam reforming behavior of methane.

It is important to note, however, that the marginal inclusion probabilities shown in Figure 3-3 do not capture correlations among the values of the reaction rate parameters or patterns of joint inclusion/exclusion. The joint posterior probability distribution of the rate parameters, which we have sampled using MCMC, in fact contains much more information. In particular, it contains information about combinations of reactions and how well particular combinations are supported by the current data.

Reaction prior probability $0.2\delta(k_{i,f}) + 0.8\mathcal{C}(k_{i,f})$		Reaction prior probability $0.5\delta(k_{i,f}) + 0.5\mathcal{C}(k_{i,f})$		Reaction prior probability $0.8\delta(k_{i,f}) + 0.2\mathcal{C}(k_{i,f})$	
Freq	Excluded pairs†	Freq	Excluded pairs†	Freq	Excluded pairs†
16157	–	521	6, 12, 14	4221	3, 5, 7, 9, 12, 14, 15
4894	7	496	6, 8, 10, 12, 14	3533	3, 5, 9, 10, 12, 14, 15
4845	6	493	7, 11	3447	3, 5, 6, 7, 11, 14, 15
4545	12	477	7, 9, 11, 14	3379	3, 5, 6, 10, 11, 14, 15
4326	9	464	3, 6, 12, 14	3340	3, 5, 7, 9, 11, 14, 15
4237	10	457	3, 6, 10, 11, 14	3318	3, 5, 6, 7, 12, 14, 15
4168	11	454	6, 10, 12, 14	3153	3, 6, 8, 10, 12, 14, 15
3954	3	451	6, 12	3033	3, 7, 8, 9, 12, 14, 15
3908	15	451	7, 11, 15	2878	3, 5, 9, 10, 11, 14, 15
3902	14	441	6, 10, 11, 15	2640	3, 7, 8, 9, 11, 14, 15

†Reaction pairs are denoted here by the number associated with the forward reaction

Table 3.6: The ten models with highest posterior probability in Example 2, for each choice of prior.

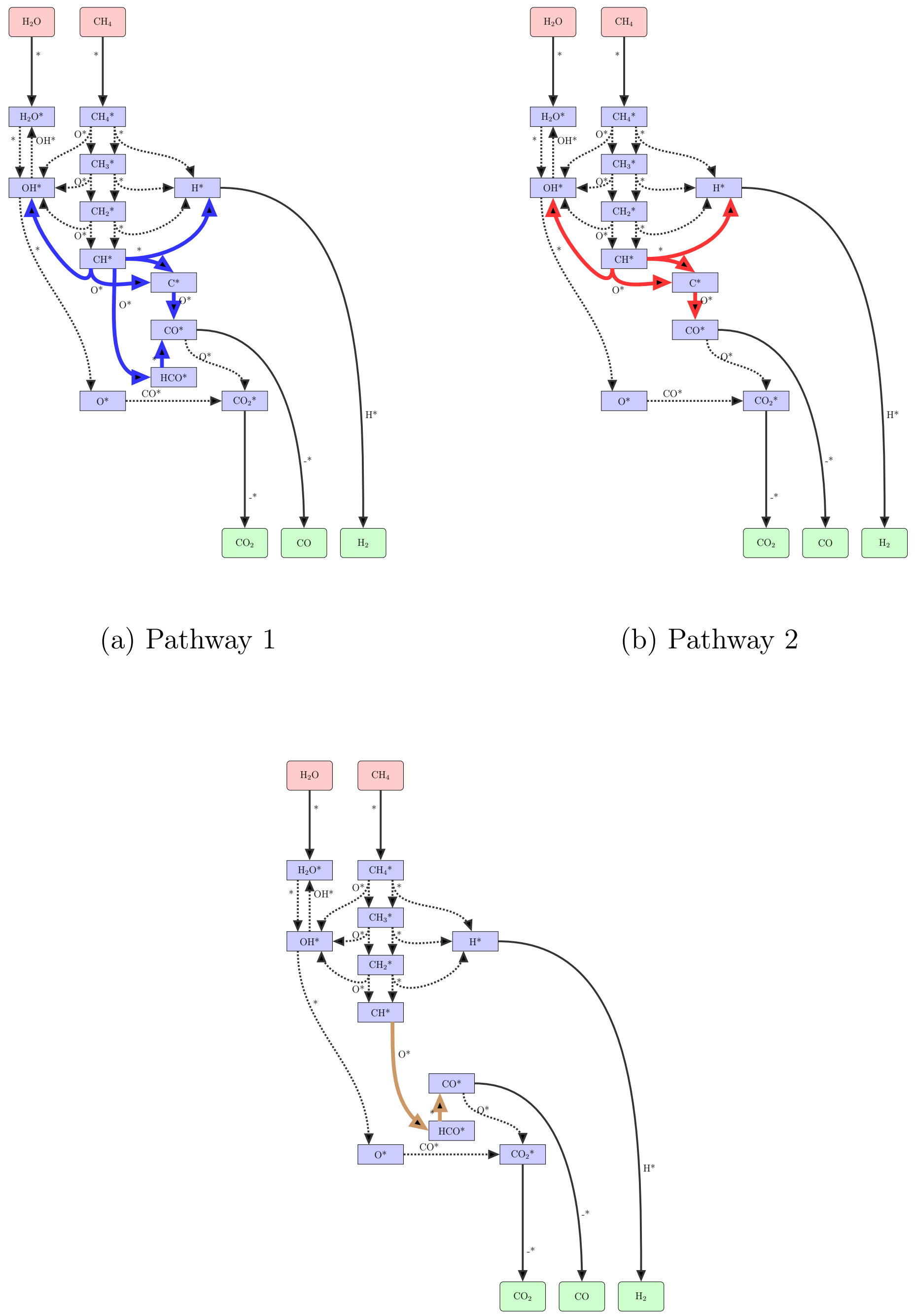


(a) Prior: $0.2\delta(k_{i,f}) + 0.8\mathcal{C}(k_{i,f})$

(b) Prior: $0.5\delta(k_{i,f}) + 0.5\mathcal{C}(k_{i,f})$

(c) Prior: $0.8\delta(k_{i,f}) + 0.2\mathcal{C}(k_{i,f})$

Figure 3-3: Posterior reaction inclusion probabilities of all reactions for the three prior specifications in Example 2. The red line indicates the prior reaction inclusion probability.



(a) Pathway 1

(b) Pathway 2

(c) Pathway 3

Figure 3-4: Reaction pathways for *steam reforming* of methane on rhodium (Example 2). Pathway 1 involves both species C* and HCO*, Pathway 2 excludes HCO*, and Pathway 3 excludes C* species. All other reactions that are treated as uncertain and that do not involve C* and HCO* are dotted. Reactions involving gas-phase species are shown as regular lines.

One way of interrogating the joint information embedded in the posterior distribution is to focus attention on particular *pathways* in the reaction network. Looking at the reaction network in Figure 3-2a (which contains *all* the proposed reactions), it is possible to discern three clear pathways for the conversion of reactants H_2O and CH_4 into products CO_2, CO, and H_2. The first pathway includes both C* and HCO* species, the second pathway excludes HCO* and retains C*, and the third pathway excludes C* but retains HCO*. The three possible reaction pathways are shown schematically in Figure 3-4. We use samples from the joint posterior distribution to quantify the degree to which each of these pathways is supported by available data. The posterior probability of each pathway is obtained by computing the fraction of posterior samples (i.e., candidate models) that contain the pathway. It is important to note that the probabilities obtained in this fashion correctly account for uncertainties in the other reactions (i.e., reactions not part of the pathway under consideration) by marginalizing over them. This contrasts with a method that simply compares three models, one corresponding to each pathway, while arbitrarily fixing or excluding all the other reactions. Given the data produced by the steam reforming experiments of McGuire et al. [86], the estimated posterior probabilities of the three pathways are shown in Table 3.7. We observe that the dominant pathway is the C* pathway. The HCO* pathway has nearly zero probability. This conclusion supports the view commonly held in the literature that steam reforming of methane operates through the C* pathway and that the inclusion of HCO* in the kinetic model is superfluous [86, 85].

Pathway	Prior 1		Prior 2		Prior 3	
	Prior	Posterior	Prior	Posterior	Prior	Posterior
1 (both C* and HCO* present)	0.536	0.643	0.177	0.249	0.026	0.039
2 (only C* present)	0.302	0.343	0.529	0.733	0.633	0.961
3 (only HCO* present)	0.162	0.010	0.294	0.012	0.340	0.000

Table 3.7: Prior and posterior pathway probabilities for steam reforming of methane, Example 2.

3.3.6 Example 3: Dry reforming of methane with real data

In the third application of our inference framework, we infer chemical kinetic models for *dry reforming* of methane using experimental data from a stagnation flow reactor reported in [85]. Operating conditions for the experiment are given in Table 3.8, and further specifics on the experimental data set (e.g., measured species and their locations) can be found in [85]. All three prior specifications discussed in Section 3.3.3 are again considered. As in the previous example, 200 000 posterior samples are simulated from a distribution encompassing all 15 uncertain reaction pairs given in Table 3.1, for each prior specification.

Table 3.9 shows the ten most probable models for each prior specification and their corresponding frequencies in 160 000 posterior samples. The highest posterior probability model obtained with Prior 1 includes all the reactions; as in the previous example, the weight specification of Prior 1 naturally favors larger models. With Prior 2, i.e., the indifference prior, the posterior excludes many reactions, slightly more than in Example 2. This reduction is again a demonstration of the penalty on model complexity built into evaluations of the marginal likelihood. The sparsity-promoting prior (Prior 3) pushes the posterior towards even smaller models, as seen the third column of Table 3.9. Reaction networks corresponding to the highest-frequency models for the three prior settings are illustrated in Figure 3-5.

Condition	Value
Inlet composition (by mole fractions)	10% CH_4 and 15% CO_2 (balance Ar)
Inlet temperature	25°C
Catalyst surface temperature	800°C
Inlet velocity	0.9 ms^{-1}
Reactor pressure	300 Torr
F_{cg}	56

Table 3.8: Experimental operating conditions for Example 3 [85].

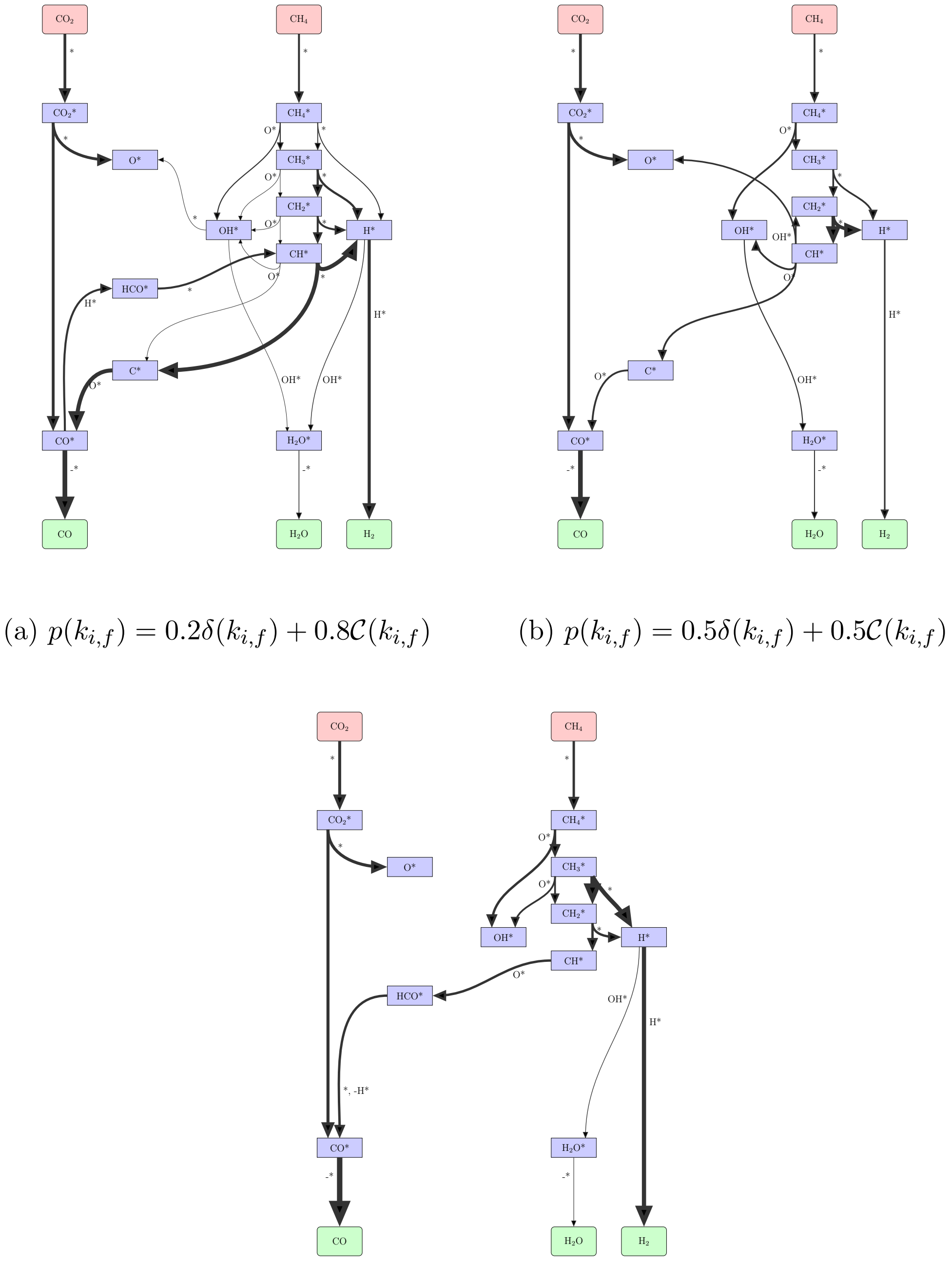
Marginal posterior inclusion probabilities of all reaction pairs for the three prior specifications are shown in Figure 3-6. We see that the posterior inclusion probabilities of all the reactions deviate from their prior inclusion probabilities, in contrast to Example 2. This suggests that the experimental data used for this dry reforming example is influenced by—and thus contains information about—every single reaction pair. As in steam reforming, the inclusion of reaction pair 4–19 with probability one confirms that the inference procedure is working well.

Reaction prior probability $0.2\delta(k_{i,f}) + 0.8\mathcal{C}(k_{i,f})$		Reaction prior probability $0.5\delta(k_{i,f}) + 0.5\mathcal{C}(k_{i,f})$		Reaction prior probability $0.8\delta(k_{i,f}) + 0.2\mathcal{C}(k_{i,f})$	
Freq	Excluded pairs†	Freq	Excluded pairs†	Freq	Excluded pairs†
7616	–	537	1, 2, 5, 9, 11, 14	4635	1, 3, 5, 10, 11, 12, 13
4950	6	488	1 10, 11, 12	3507	1, 2, 9, 10, 11, 14, 15
4800	5	484	1, 5, 10, 11, 14	3314	3, 5, 6, 10, 11, 12, 13
2892	7	445	1, 2, 5, 11, 14	3245	1, 2, 5, 9, 10, 11, 12, 13
2559	2, 6	401	1, 2, 5, 11	2608	1, 2, 9, 10, 11, 12, 13
2393	1	396	5, 6, 7, 14	2428	1, 2, 9, 10, 12, 14, 15
2107	12	392	5, 6, 7, 11, 14	2370	3, 5, 9, 10, 11, 12, 13
2023	15	391	5, 6, 10, 12, 15	1607	1, 3, 5, 7, 12, 14, 15
1889	9	369	5, 6, 7, 11	1511	3, 5, 6, 7, 12, 14, 15
1884	14	365	5, 6, 7, 14, 15	1448	1, 3, 7, 8, 12, 14, 15

†Reaction pairs are denoted here by the number associated with the forward reaction

Table 3.9: The ten models with highest posterior probability in Example 3, for each choice of prior.

We also compute the posterior probabilities of the three distinct pathways shown in Figure 3-7. Pathway 1 includes both C* and HCO*, pathway 2 excludes HCO* and retains C*, and pathway 3 excludes C* and retains HCO*. The posterior probabilities of the three pathways are shown in Table 3.10. Compared to the corresponding results for steam reforming (Table 3.7), the present results suggest that the HCO* pathway



(a) $p(k_{i,f}) = 0.2\delta(k_{i,f}) + 0.8\mathcal{C}(k_{i,f})$

(b) $p(k_{i,f}) = 0.5\delta(k_{i,f}) + 0.5\mathcal{C}(k_{i,f})$

(c) $p(k_{i,f}) = 0.8\delta(k_{i,f}) + 0.2\mathcal{C}(k_{i,f})$

Figure 3-5: Reaction networks of the highest posterior probability models for dry reforming of methane (Example 3), under different prior specifications. Edge thicknesses are proportional to reaction rates calculated using posterior mean values of the rate parameters.

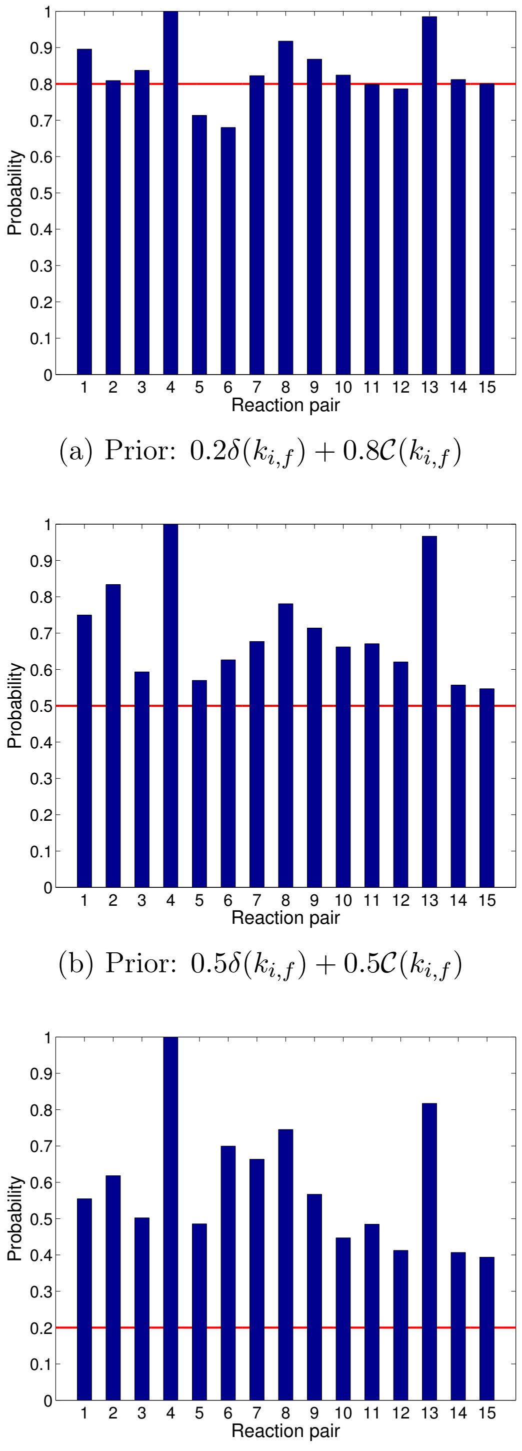
is not unimportant to dry reforming. In other words, it is possible that dry reforming of methane is realized through the generation of HCO*. With an indifference prior, the HCO*-only pathway has a posterior probability of 8%. With a sparsity promoting prior, the posterior probability of pathway 1 decreases dramatically and the posterior places 23% of its mass on the HCO*-only route. That said, the C* pathway remains very much the dominant pathway given the current data. Pathway 1 also has strong support except in the case of the sparsity-promoting prior, which effectively forces the posterior to "choose" between the two more parsimonious options. A clearer conclusion can only result from collecting more data and repeating this analysis.

Pathway	Prior 1		Prior 2		Prior 3	
	Prior	Posterior	Prior	Posterior	Prior	Posterior
1 (both C* and HCO* present)	0.536	0.596	0.177	0.244	0.026	0.034
2 (only C* present)	0.302	0.347	0.529	0.677	0.633	0.732
3 (only HCO* present)	0.162	0.056	0.294	0.080	0.340	0.234

Table 3.10: Posterior pathway probabilities for dry reforming of methane, Example 3.

3.3.7 Efficiency of posterior sampling

To verify the numerical results reported in the preceding sections, we performed three independent MCMC runs for each example problem and each prior specification, with different initial guesses for the rate parameters $\tilde{k}$ in each case. Overall, the three replicate runs yielded very similar results; the independent chains were able to identify the high posterior probability models and accurately reproduce their probabilities. Yet the quality of these posterior estimates, of course, depends on the number of posterior samples employed—i.e., the length of the MCMC chains. Because the forward models $\boldsymbol{G}(\tilde{k}, \hat{k})$ in this setting are computationally expensive, it is practically important to

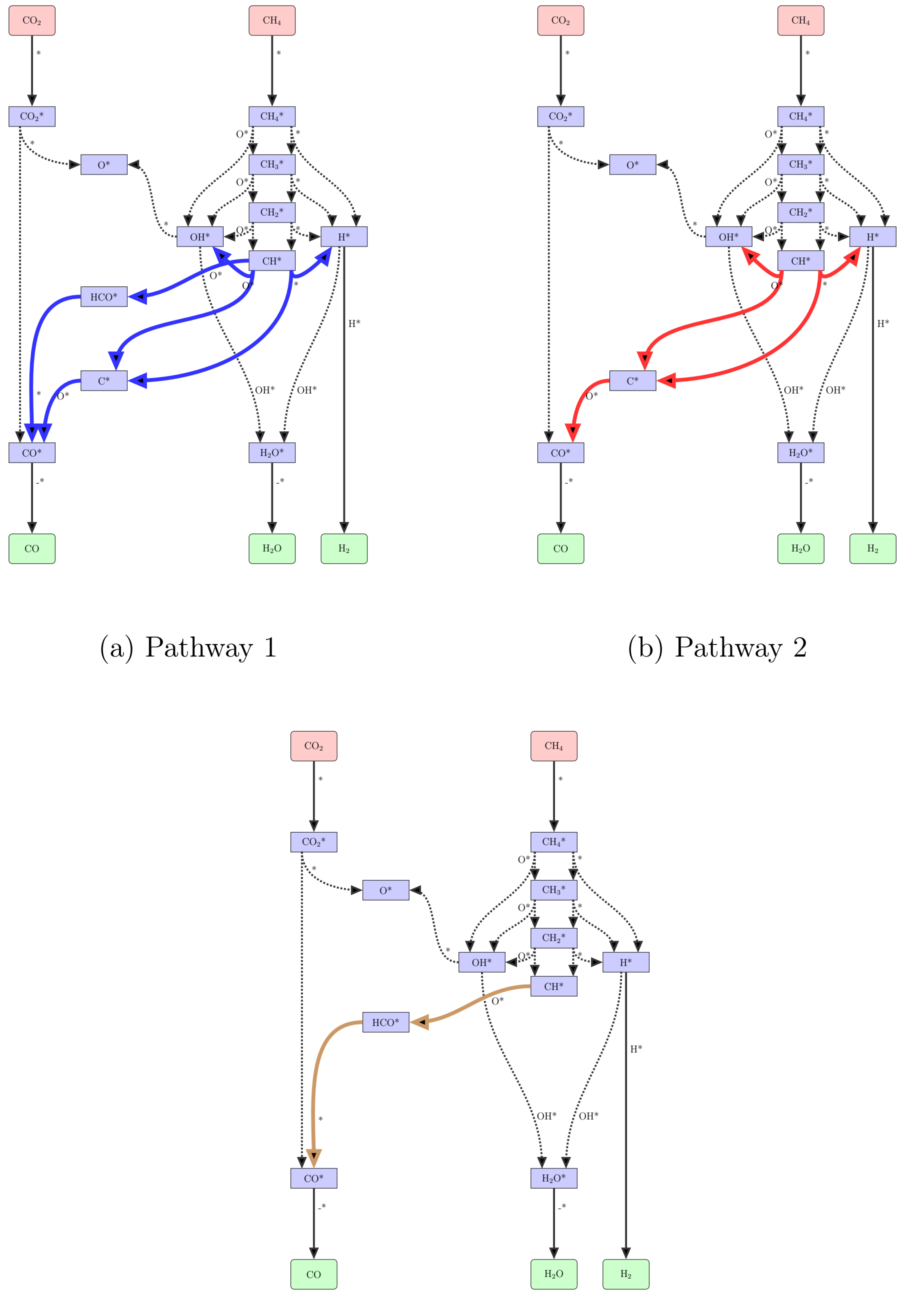


(a) Prior: $0.2\delta(k_{i,f}) + 0.8\mathcal{C}(k_{i,f})$

(b) Prior: $0.5\delta(k_{i,f}) + 0.5\mathcal{C}(k_{i,f})$

(c) Prior: $0.8\delta(k_{i,f}) + 0.2\mathcal{C}(k_{i,f})$

Figure 3-6: Posterior reaction inclusion probabilities of all reactions for the three prior specifications in Example 3. The red line indicates the prior reaction inclusion probability.

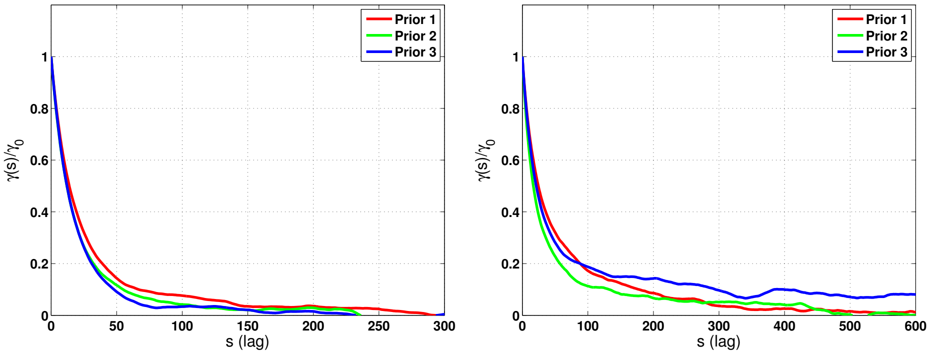


(a) Pathway 1

(b) Pathway 2

(c) Pathway 3

Figure 3-7: Reaction pathways for *dry reforming* of methane on rhodium (Example 3). Pathway 1 involves both species C* and HCO*, Pathway 2 excludes HCO*, and Pathway 3 excludes C* species. All other reactions that are treated as uncertain and that do not involve C* and HCO* are dotted. Reactions involving gas-phase species are shown as regular lines.



(a) Example 2

(b) Example 3

Figure 3-8: Autocorrelation at lag s of the log-posterior of the MCMC chains.

limit the number of samples. As described in Section 3.2, the variance of a posterior estimate for a fixed number of samples depends on how well the chain is mixing [3, 105]. The adaptive MCMC scheme employed here has been shown to significantly improve mixing over non-adaptive schemes [73], but it is nonetheless important to assess the quality of its sampling.

A useful diagnostic for the quality of MCMC mixing is the empirical autocorrelation of the chain. In particular, we compute the correlation between samples as function of lag time. A steep decay in this autocorrelation means that successive samples are less correlated and more nearly independent. While one could compute the empirical autocorrelation for each reaction parameter individually, we instead summarize MCMC mixing by computing the autocorrelation of successive values of the log-posterior density. This is reported in Figure 3-8, for Examples 2 and 3 with all three prior specifications.

The decay of the autocorrelation is relatively good in both cases, though the MCMC chains mix more quickly in Example 2 than in Example 3. This difference can be ascribed to differences in posterior structure. In Example 2, the C* pathway is largely dominant, while in the dry reforming case of Example 3, both the C* and HCO* pathways have appreciable posterior probabilities. The MCMC chain in Example 3 thus has to switch between pathways more frequently, and each switch requires the inclusion and exclusion of multiple reactions. Even with the present adaptive proposal distribution, this coordinated inclusion and exclusion is a relatively "large" jump in the model space. Thus, while mixing is adequate in the current example, a more computationally efficient approach—i.e., one that could achieve similar results with fewer samples—might involve correlated proposal mechanisms that can learn not just the marginal structure of the posterior in each parameter direction but the joint structure of posterior. The design of such proposal mechanisms is a topic of ongoing research.

3.3.8 Posterior parameter uncertainties

Thus far, we have focused our analysis on the posterior description of uncertainties in *model structure*. But the across-model Bayesian inference framework also automatically produces a full description of uncertainties in rate *parameter values*. In other words, for every model in the posterior distribution, MCMC samples describe the joint probability distribution of the rate parameters that are included in that model (i.e., that are non-zero). Quantifying these parameter uncertainties is important when developing a rigorous assessment of uncertainties in model predictions.

Here we provide one example of the posterior parameter uncertainties obtained using our inference framework. Figure 3-9 shows 1-D and 2-D marginal distributions of the rate constants of the highest-probability model for steam reforming (Example 2), using Prior 3. This model includes 8 of the 15 possible reactions (as described in Table 3.6), and thus the continuous distribution over rate parameter values is supported on an eight-dimensional space. The diagonal of Figure 3-9 shows the marginal probability density function of one parameter at a time, while the off-diagonal elements show the joint probability density of each pair of parameters (marginalizing out the other six). The prior probability density was uniform over each box, and thus the more focused contours indicate a significant reduction in prior uncertainty. There is also some complex (and certainly non-Gaussian) structure in each pairwise marginal. This is the uncertainty given the experimental data at hand. Further reduction in parameter uncertainty would require the injection of additional data into the inference process.

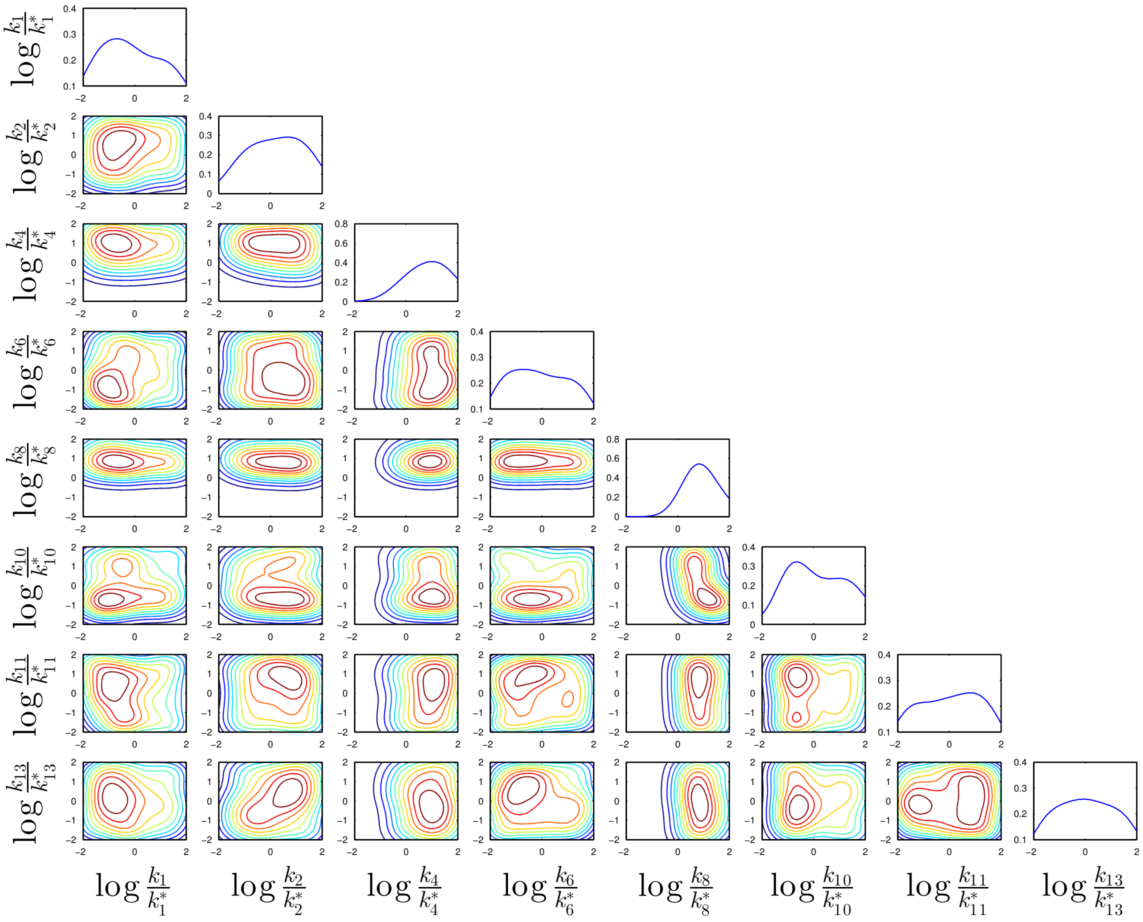


Figure 3-9: 1-D and 2-D posterior marginals of the rate constants of the highest-posterior-probability model for steam reforming (from Example 2), beginning with the prior $p(k_{i,f}) = 0.8\delta(k_{i,f}) + 0.2\mathcal{C}(k_{i,f})$. The logarithms here are base 10.

Chapter 4

Network-aware inference

The network interaction of species in reaction models induces a special structure to the network inference problem. The production/destruction of a species in a chemical reaction model is directly linked to the concentrations of other species with which it reacts. Therefore, the rate of production/consumption of a species is necessarily zero if those other species are absent from the system. Practically, this means that by excluding a reaction from a network, many other reactions are effectively eliminated (i.e., have zero reaction rate). The rate constants of reactions whose rate is zero do not affect the likelihood function and thus cannot be informed by the available data. Another common difficulty with network inference problems is that the available data is often sparse—data are directly linked to only a few chemical species. For example, in applications of catalytic chemistry, only gas-phase species measurements are practically feasible. While inferring protein signalling networks, only a few protein concentrations may be measurable. The sparsity of data can mean that in spite of a reaction being active (i.e., having non-zero rate), it has no influence on the observables. Again, the rate constant values of these reactions are not informed by data. In this chapter, we present methods that exploit the *network-based* species interactions to improve the sampling efficiency of across-model network samplers.

The central theme of our network-aware MCMC methods is that the network structure of species interactions can be exploited to engineer improved between-model parameter proposals. Firstly, we analyze the reaction network given a set of proposed reactions to identify reactions that actually impact the observables. In other words, we identify the smallest subset of the proposed reactions that would produce an identical value of the marginal likelihood as the proposed set of reactions. This information is then incorporated in the design of improved proposal distributions. Identification of networks with identical marginal likelihood further allows variance reduction through analytical computation of some conditional expectations. The second contribution of this chapter is due to the recognition that between-model proposals can benefit from designing better move types. For example, proposing good moves between two networks in many instances requires updating even the rate constants of reactions present in both networks. We present a method to identify "key" reactions whose rate constants are also included in the between-model parameter proposals. This step is then combined with the first idea to further enhance sampling efficiency.

4.1 Chemical reaction network structure

4.1.1 Reaction network elements

A chemical reaction network generally consists of two different elements. Chemical species S and the interaction between species given by reactions R. Consider a simple reaction network shown schematically in Figure 4-1. The reaction network consists of 6 nodes denoting the chemical species and 6 edges corresponding to reactions. An alternative representation of the reaction network involves writing the list of all reactions:

1. Reaction 1: $\textcircled{1} \rightarrow \textcircled{2}$

2. Reaction 2: $\textcircled{2} \rightarrow \textcircled{3}$

3. Reaction 3: $\textcircled{3} \rightarrow \textcircled{4}$

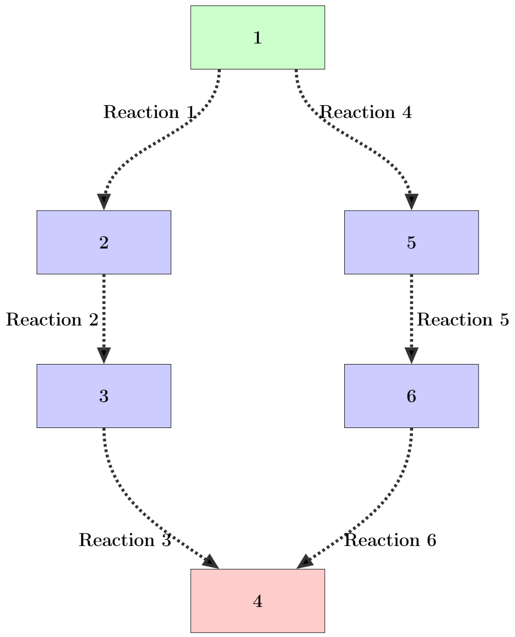


Figure 4-1: A simple reaction network

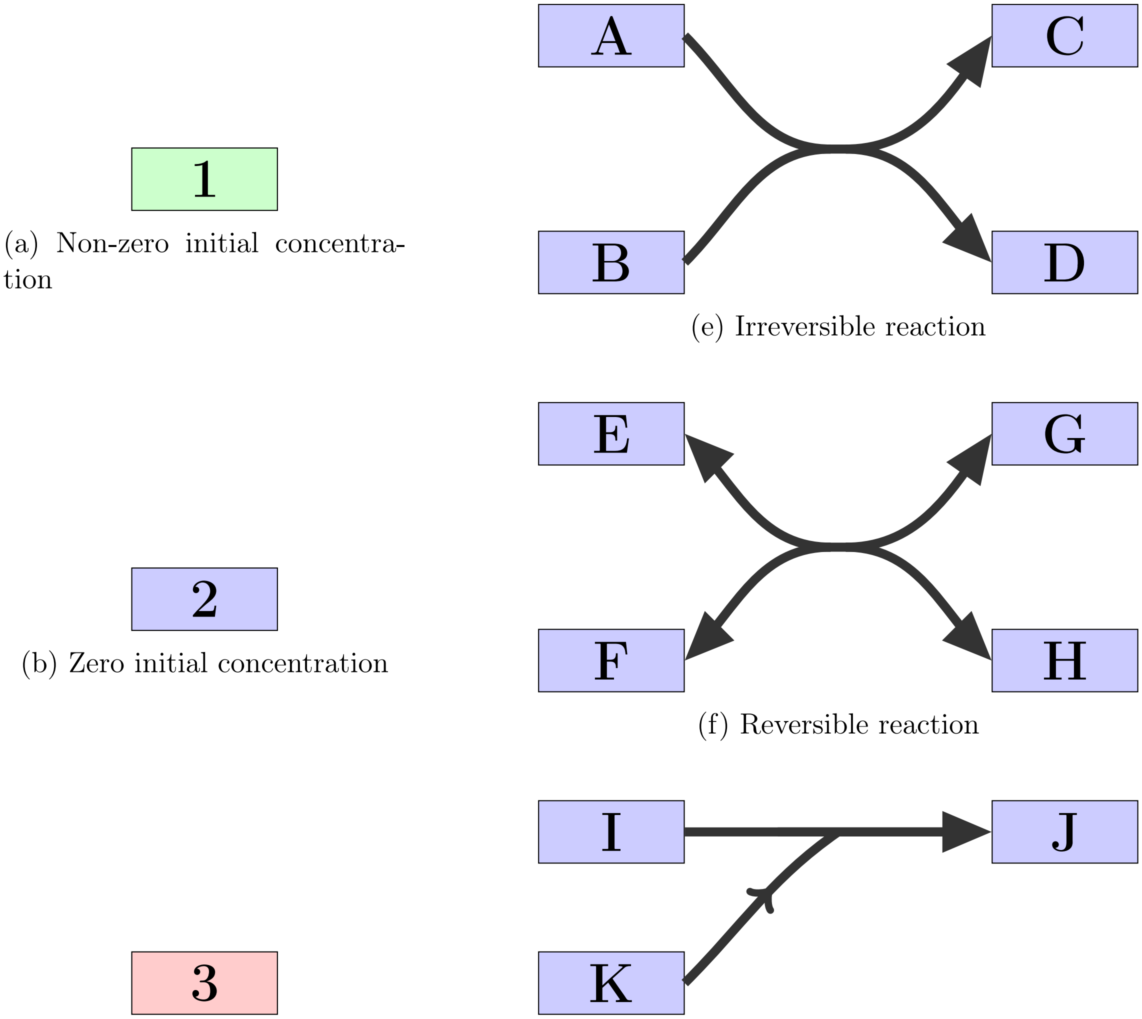
4. Reaction 4: ① → ⑤

5. Reaction 5: ⑤ → ⑥

6. Reaction 6: ⑥ → ④

From a data-analytic perspective, we will classify all species into three categories. Species initially present in the data-generating system are shown in green (Figure 4-2a), species produced only during the operation of the system are shown in blue (Figure 4-2b), and species that are either directly observed, or are directly linked to the observed data—referred to as *observables*—are shown by red nodes (Figure 4-2c). Next we discuss some common species interaction elements (reactions) present in reaction networks.

Figure 4-2e depicts an irreversible reaction in which Ⓐ and Ⓑ are the reactants and Ⓒ and Ⓓ are the products. Figure 4-2f shows a reversible reaction involving with Ⓔ and Ⓕ the reactants and Ⓖ and Ⓗ the products. Figure 4-2g denotes



(a) Non-zero initial concentration

(b) Zero initial concentration

(c) Observed in experiments

(d) Species

(e) Irreversible reaction

(f) Reversible reaction

(g) Reaction with enzyme/catalyst I

(h) Reactions

Figure 4-2: Common reaction network elements

an irreversible reaction between $\textcircled{K}$ and $\textcircled{J}$, with $\textcircled{I}$ acting as an enzyme/catalyst. Enzymes are chemical species that are needed for the reaction to proceed, but do not get consumed or produced during the course of the reaction. All species interactions (reactions) are defined by ordinary differential equations, with the corresponding rate expressions obtained by the law of mass action or Michaelis-Menten kinetics.

4.1.2 Effective reaction network

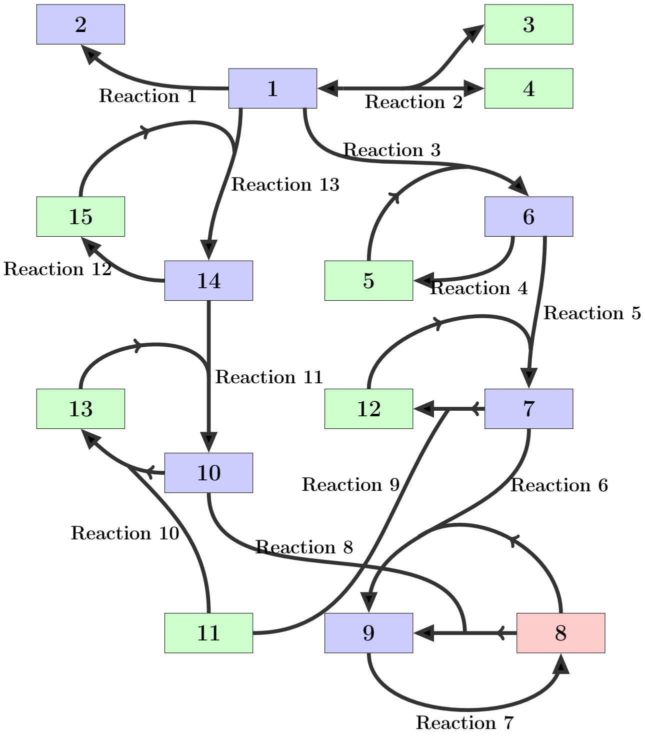
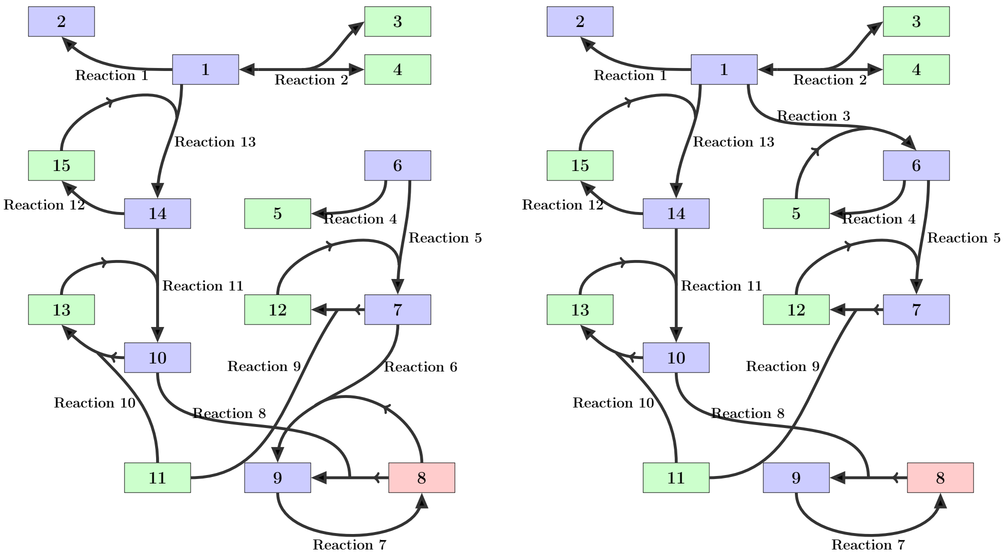


Figure 4-3: Reaction network with all reactions

A reaction network may contain reactions that may not be active or reactions that are active and yet incapable of impacting the observables due to the network-based interactions of all species. Consider, for example, a set of proposed reactions given by the full reaction network in Figure 4-3. Two possible reduced reaction networks, obtained by removing reactions 3 and 6, respectively, from the full reaction network 4-3 are shown in Figure 4-4. We define the *effective network* of a reaction network to be the smallest subset of all reactions in the network that produces an identical value of the observables as the given reaction network. This implies that the additional reactions in a network compared to the effective network do not affect the observable value for any parameter setting, and the reaction network has the same marginal likelihood value as the effective reaction network. Both reduced networks shown in Figure 4-4 have the same effective network (Figure 4-5). In reaction network 1, the non-production of species 6 renders reactions 4, 5, 6, and 9 inactive, and thus the value of the observable ⑧ is independent of their rate constant values. In case of reaction network 2, although reactions 3, 4, 5, and 9 are active, they are linked to the observable ⑧ through species ① and ⑪, which are enzymes. Recall that catalyst/enzyme concentration is not affected by the reaction in which it participates. Thus, the observable is again independent of the rate constants of reactions 3,4,5, and 9. Given a set of proposed reactions, one can obtain all the plausible reaction networks and their corresponding effective networks.

4.1.3 Determining effective networks from proposed reactions

Before we begin sampling over the space of models and parameters, we first determine the effective networks of all plausible networks. If N is the total number of proposed reactions, the set of possible networks may be 2^N, although incorporating prior knowledge to eliminate highly unlikely models may also be a practical choice. In either case, if the number of possible networks is very high, one may choose to determine effective networks online only for models visited by the sampler. Our procedure to determine the effective network given a set of reactions and observables is given by Algorithm 5. The approach we employ to determine the effective network



(a) Reaction network 1

(b) Reaction network 2

Figure 4-4: Two reaction networks with the same effective network

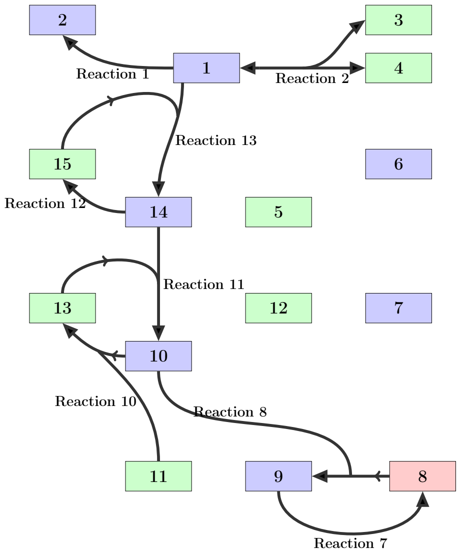


Figure 4-5: Effective reaction network

involves first identifying all reactions that are active given all species with nonzero initial concentration and then testing all active reactions to check if they actually influence the observables. The steps involved in checking whether a particular active reaction influences the observables are given by Algorithm 6.

4.1.4 The space of model clusters

Given a set of proposed reactions, we can determine the effective reaction networks for all reaction networks using the algorithm in the last section. We are now in a position to define *clusters of models*. A cluster is defined as the collection of all models with the same effective network. Thus, assuming the models have been assigned a prior distribution $p(\{M_j\})$, the cluster prior probability is given by

$$p(C_K) = \sum_{M_m \in C_K} p(M_m). \tag{4.1}$$

Further, the set of clusters has the following property:

$$C_K \cap C_J = \emptyset \text{ for } K \neq J \tag{4.2}$$

and

$$\bigcup_K C_K = \mathcal{M}, \tag{4.3}$$

where $\mathcal{M}$ is the complete space of all networks.

4.2 Reversible jump Markov chain Monte Carlo

In Chapter 3, we described a fixed-dimensional adaptive MCMC algorithm for model-space sampling. Here we work in the more general reversible-jump MCMC (RJMCMC) framework which affords greater flexibility in proposal constructions. However,

Algorithm 5 Effective reaction network from a set of reactions

```
Given: $\boldsymbol{R_{prop}}$: proposed reactions; $\boldsymbol{S_{in}}$ species initially present;
$\boldsymbol{R_e}$: reactions in effective network, $\boldsymbol{R_e} = \varnothing$; $\boldsymbol{S_e}$: species in the effective network, $\boldsymbol{S_e} = \boldsymbol{S_{in}}$
$\boldsymbol{r_i}$: reactants of reaction $i$; $\boldsymbol{p_i}$: products of reaction $i$; $\boldsymbol{a_i}$: enzymes of reaction $i$
$n_e^{'} = 0$, $t_e^{'} = 0$
while $n_e^{'} \neq |\boldsymbol{R_e}|$ and $t_e^{'} \neq |\boldsymbol{S_e}|$ do
    $n_e^{'} = |\boldsymbol{R_e}|$ and $t_e^{'} = |\boldsymbol{S_e}|$
    for $i = 1$ to $|\boldsymbol{R_{prop}}|$ do
        if Reaction $R_i$ irreversible then
            if $(\boldsymbol{r_i} \cup \boldsymbol{a_i}) \in \boldsymbol{S_e}$ then
                $\boldsymbol{R_e} = \boldsymbol{R_e} \cup R_i$ and $\boldsymbol{S_e} = \boldsymbol{S_e} \cup \boldsymbol{p_i}$
            end if
        else if Reaction $R_i$ is reversible then
            if $(\boldsymbol{r_i} \cup \boldsymbol{a_i}) \in \boldsymbol{S_e}$ or $(\boldsymbol{p_i} \cup \boldsymbol{a_i}) \in \boldsymbol{S_e}$ then
                $\boldsymbol{R_e} = \boldsymbol{R_e} \cup R_i$ and $\boldsymbol{S_e} = \boldsymbol{S_e} \cup \boldsymbol{p_i} \cup \boldsymbol{r_i}$
            end if
        end if
    end for
end while
$\boldsymbol{R_{active}} = \boldsymbol{R_e}$
for $i = 1$ to $|\boldsymbol{R_e}|$ do
    Inf $\leftarrow$ Algorithm 6 ($\boldsymbol{R_{active}}$, $R_i$)
    if Inf==0 then
        $\boldsymbol{R_e} = \boldsymbol{R_e} \setminus \{R_i\}$
    end if
end for
```

the ideas discussed here can be applied to the adaptive MCMC framework too. Recall that the reversible jump MCMC is a general across-model sampling algorithm that jointly samples the space of models and their corresponding parameters (Section 2.6.4).

4.2.1 Parameter proposals for RJMCMC

The sampling efficiency of reversible jump MCMC simulation hinges on the choice of the map function $\boldsymbol{f}$, the parameter proposal distributions $q(\boldsymbol{u}|\boldsymbol{k}_M)$ and $q(\boldsymbol{u'}|\boldsymbol{k}_{M'})$, and model-move proposal distribution $q(M'|M)$. The model-move proposal $q(M'|M)$

Algorithm 6 Algorithm to check if reaction R_I influences the observables

```
Given: Active reactions $\boldsymbol{R_{act}}$; Observables $\boldsymbol{O}$
$\boldsymbol{S_{inc}}$: collection of species, $\boldsymbol{S_{inc}} = \boldsymbol{r_i} \cup \boldsymbol{p_i}$; $\boldsymbol{R_{inc}}$: collection of reactions, $\boldsymbol{R_{inc}} = R_i$
$t^{'}_{inc} = 0$
while $t^{'}_{inc} \neq |\boldsymbol{S_{inc}}|$ do
    $t^{'}_{inc} = |\boldsymbol{S_{inc}}|$
    for $j = 1$ to $|\boldsymbol{R_{act}}|$ do
        if $R_j \notin \boldsymbol{R_{inc}}$ then
            if $\boldsymbol{r_j} \cup \boldsymbol{a_j} \in \boldsymbol{S_{inc}}$ or $\boldsymbol{p_j} \in \boldsymbol{S_{inc}}$ then
                $\boldsymbol{R_{inc}} = \boldsymbol{R_{inc}} \cup R_j$
            end if
        end if
    end for
    for $j = 1$ to $|\boldsymbol{R_{inc}}|$ do
        $\boldsymbol{S_{inc}} = \boldsymbol{S_{inc}} \cup \boldsymbol{p_j} \cup \boldsymbol{r_j}$
    end for
end while
if $\boldsymbol{O} \in \boldsymbol{S_{inc}}$ then
    Inf=1
end if
```

is generally chosen so that every move adds or delete one reaction. This choice is made due to the difficulty in constructing effective parameter proposals in high dimensions. The selection of the jump function $\boldsymbol{f}$ and the parameter proposals q is based on the goal of improving between-model acceptances for both the forward ($M \to M'$) and reverse ($M' \to M$) model moves. Higher between-model acceptance rates may be obtained by "aligning" densities between the posterior and the proposals corresponding to the two models between which moves are proposed. As an example, consider moves between a one-dimensional model M_1 (unnormalized posterior density: $\tilde{p}(k_1|\boldsymbol{\mathcal{D}})$) and a two-dimensional model M_2 (unnormalized posterior density: $\tilde{p}(k_{2,1}, k_{2,2}|\boldsymbol{\mathcal{D}})$) shown in Figure 4-6, accomplished with proposal $q(u|k_1)$. By choosing the function f and the shape of the proposal $q(u|k_1)$ such that the regions of high density and low density in the two spaces ((k_1, u) and $(k_{2,1}, k_{2,2})$) are mapping to each other, respectively, and the joint densities $\tilde{p}(k_1|\boldsymbol{\mathcal{D}})q(u|k_1)$ and $\tilde{p}(f(k_{2,1}, k_{2,2})|\boldsymbol{\mathcal{D}})$ are similar in value in the two spaces, high between-model acceptance rates may be achieved. Intuitively, the above construction is attempting to choose $\boldsymbol{f}$ and q so as to make the acceptance rate close to 1 for all moves between the two spaces.

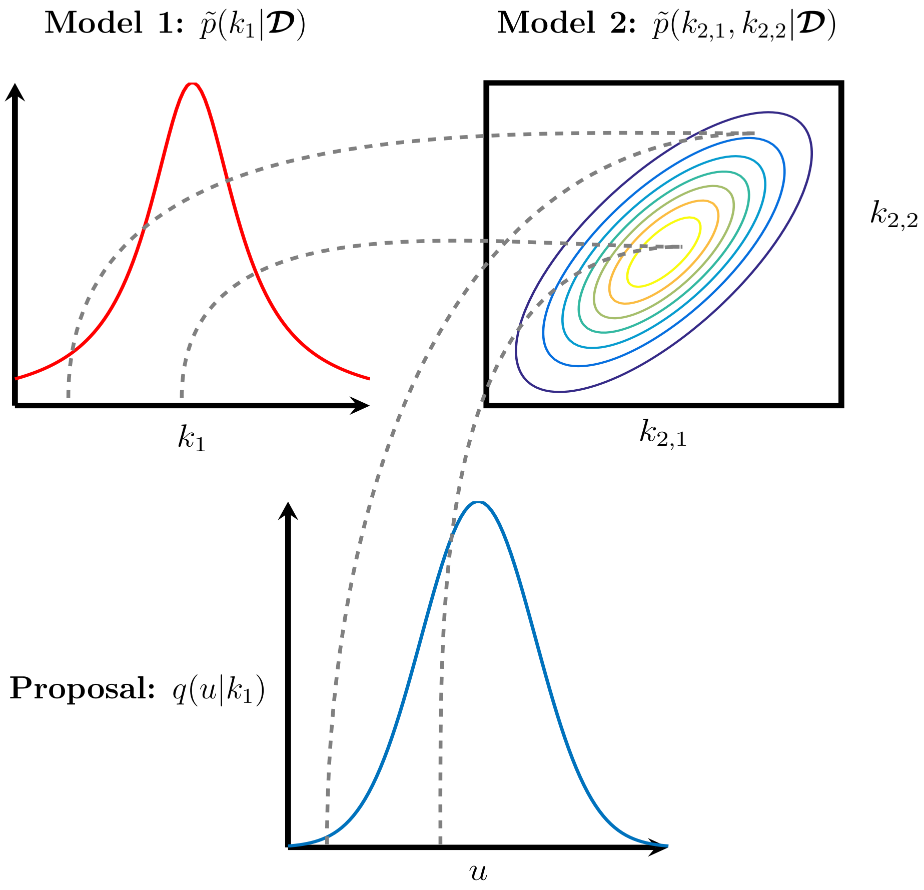

Model 1: $\tilde{p}(k_1|\boldsymbol{\mathcal{D}})$
Model 2: $\tilde{p}(k_{2,1}, k_{2,2}|\boldsymbol{\mathcal{D}})$
$k_{2,2}$
k_1
$k_{2,1}$
Proposal: $q(u|k_1)$
u

Figure 4-6: Efficient RJMCMC: align densities on (k_1, u) to $(k_{2,1}, k_{2,2})$ accurately

The selection of a good map $\boldsymbol{f}$ and the design of proposal distribution $q(\boldsymbol{u}|\boldsymbol{k}_M)$ in RJMCMC is challenging and often chosen based on pilot simulations. The high cost and typically poor performance of the pilot-run based RJMCMC has prompted the development of methods for automatic proposal construction [1, 20, 36, 38, 58, 59]. All the above methods attempt to increase the acceptance rate of between-model moves at an additional computational expense, and have shown to improve performance in a number of cases. Brooks et al. [20] provide a general framework to understand and construct efficient reversible-jump proposals based on an analysis of the acceptance probability. Our work fits in their their n^{th}-order condition proposal framework and we briefly review it here.

Centered n^{th}-order condition based proposals

The n^{th}-order ($n \geq 1$) proposal conditions of Brooks et al. [20] is based on setting a series of derivatives (with respect to $\boldsymbol{u}$) of the acceptance ratio A (2.37) for proposal moves between models M and M' to the zero vector at a specific point $\boldsymbol{c}_{M \to M'}(\boldsymbol{k}_M)$ known as the *centering point*:

$$\nabla^n A[(M, \boldsymbol{k}_M), (M', \boldsymbol{c}_{M \to M'}(\boldsymbol{k}_M))] = \mathbf{0} \tag{4.4}$$

The centering point $\boldsymbol{c}_{M \to M'}(\boldsymbol{k}_M)$ is taken to be the equivalent point of parameter vector $\boldsymbol{k}_M$ in model M'. Centering the proposal $q(\boldsymbol{u}|\boldsymbol{k}_M)$ at the conditional maximum of the posterior density $p(\boldsymbol{f}(\boldsymbol{k}_M, \boldsymbol{u}), M')$ is one intuitive choice and aims to increase the frequency of moves between models. In addition to the n^{th}-order condition, Brooks et al. [20] also introduce the zeroth-order condition in which the acceptance ratio is set to 1 at the centering point. Note the traditional random-walk Metropolis algorithm for posterior distribution on $\mathbb{R}^m$ satisfies the zeroth-order condition at a central move corresponding to a step of size $\mathbf{0}$. Similarly, Langevin algorithms satisfy both the zeroth-order and first-order conditions at the above central move. The zeroth and n^{th} order conditions aim to adapt proposal parameters on the current state of the chain $(M, \boldsymbol{k}_M)$, instead of relying on constant proposal parameters for all state transitions. Brooks et al. [20] further show that for a simple two model case, the n^{th}-order conditions are optimal in terms of the capacitance of the algorithm [80].

4.3 Network analysis for improved sampling efficiency

We now explain how we use the effective reaction networks obtained using our Algorithm 5 to design more efficient across-model samplers. We propose two methods for performance improvement. First, we use network information to determine improved parameter proposal densities $q(\boldsymbol{u}|\boldsymbol{k}_M)$. Secondly, we demonstrate how one may use sensitivity of observables to individual reactions to choose efficient between-model moves. The above network-aware parameter proposals are then used along with the sensitivity-based move types. We end by explaining how the analysis of network structure can be used to derandomize conditional expectation calculations, leading to further variance reduction.

4.3.1 Constructing parameter proposals

For nested models, as is the case in the reaction network inference problem, a natural choice of the jump function $\boldsymbol{f}$ is to choose the identity function. Thus, when proposing a move from a lower-dimensional model M to a higher dimensional model M', the rate constants of the newly added reactions is proposed according to $q(\boldsymbol{u}|\boldsymbol{k}_M)$ and the values of the rate constants of reactions common to the two models are kept fixed (henceforth $1:i$). Therefore,

$$\boldsymbol{f} := (\boldsymbol{k}_{M'}^{\boldsymbol{1:i}}, \boldsymbol{k}_{M'}^{\boldsymbol{1:a}}) = (\boldsymbol{k}_{M}^{\boldsymbol{1:i}}, \boldsymbol{u}_{M}^{\boldsymbol{1:a}}). \tag{4.5}$$

and the acceptance probability is given by

$$\alpha(\boldsymbol{k}_M, \boldsymbol{k}_{M'}) = \min\left\{1, \frac{p(M', \boldsymbol{k}_{M'}|\boldsymbol{\mathcal{D}})q(M|M')}{p(M, \boldsymbol{k}_M|\boldsymbol{\mathcal{D}})q(M'|M)q(\boldsymbol{u}|\boldsymbol{k}_M)}\right\}. \tag{4.6}$$

The reverse move in this case is deterministic. Let the proposal $q(\boldsymbol{u}|\boldsymbol{k}_M)$ be given by

$$q(\boldsymbol{u}|\boldsymbol{k}_M) = \mathcal{N}(\boldsymbol{u}; \boldsymbol{\mu}, \boldsymbol{\Sigma}). \tag{4.7}$$

To improve the chance of proposal acceptance we center the proposal distribution at the conditional mean of the posterior distribution. Next, we construct an approximation to the posterior distribution by setting the covariance of the Gaussian to be the Hessian of the conditional posterior density. In other words, we construct a Gaussian approximation to the conditional posterior distribution. In the framework of Brooks et al. [20], the above construction is equivalent to the centered second-order conditions. In the scheme described above, the mean vector $\boldsymbol{\mu}$ is set to the conditional maximum:

$$\boldsymbol{\mu} = \arg\max_{\boldsymbol{u}} p(M', (\boldsymbol{k}_M, \boldsymbol{u})|\boldsymbol{\mathcal{D}}). \tag{4.8}$$

A proposal centered at the posterior conditional maximum satisfies the first order condition:

$$\begin{aligned}\nabla \log A(M, \boldsymbol{k}_M \to M', \boldsymbol{k}_{M'})\big|_{\boldsymbol{\mu}} &= \nabla \left[\log \mathcal{L}(\boldsymbol{\mathcal{D}}; \boldsymbol{k}_M, \boldsymbol{u}) + \log p(\boldsymbol{k}_M, \boldsymbol{u}) - \log q(\boldsymbol{u}; \boldsymbol{\mu}, \boldsymbol{\Sigma})\right]\big|_{\boldsymbol{\mu}} \\ &= \boldsymbol{0}. \end{aligned} \tag{4.9}$$

Further, setting the second-derivative of the acceptance ratio at the conditional maximum $\boldsymbol{0}$, we obtain the second order condition as:

$$\begin{aligned}\nabla^2 \log A((M, \boldsymbol{k}_M) \to (M', \boldsymbol{k}_{M'})) &= \nabla^2 \left[\log \mathcal{L}(\boldsymbol{\mathcal{D}}|\boldsymbol{k}_M, \boldsymbol{u}) + \log p(\boldsymbol{k}_M, \boldsymbol{u}) - \log q(\boldsymbol{u}; \boldsymbol{\mu}, \boldsymbol{\Sigma})\right] \\ &= \boldsymbol{0}. \end{aligned} \tag{4.10}$$

Taking $\boldsymbol{\mathcal{H}}$ to be the Hessian of the conditional posterior density at $\boldsymbol{\mu}$, (4.10) yields

$$\boldsymbol{\mathcal{H}}\big|_{\boldsymbol{\mu}} + \boldsymbol{\Sigma}^{-1} = \boldsymbol{0} \implies \boldsymbol{\Sigma} = -\boldsymbol{\mathcal{H}}^{-1}\big|_{\boldsymbol{\mu}}. \tag{4.11}$$

4.3.2 Network-aware parameter proposals

The above proposal construction for between-model moves in which the parameter proposals adapt to conditional posterior densities typically lead to improved reversible jump simulations [20, 36, 38, 52]. Effectively, the above proposal construction is attempting to increase the chance of proposed moves between models to get accepted. In addition, with the first and second order conditions, the idea is to make acceptance ratio uniformly high for all transitions. However, the direct application of the centered second-order conditions for between-model moves in the context of reaction network inference has a major drawback. As discussed earlier, many reaction networks can have the same effective network. In such a case, if the proposed move is between two networks with the same effective network (i.e., the two networks belong to the same cluster), the parameter proposal adapts to the prior distribution of the newly added reaction (Figure 4-7). We propose a network-aware approach in which, because we have determined the effective networks, we design parameter proposals that adapt to the difference between the effective networks of the two networks. When the proposed move is between two networks belonging to different clusters, we construct a proposal that approximates the conditional posterior distribution of the rate constants of all reactions *not included* in the two effective networks.

Formally, suppose that the sampler proposes a move from a lower-dimensional model M to a higher-dimensional model M'. Let the effective networks of the two models M and M' be M_e and M'_e, respectively. Following our choice of the proposal $q(M'|M)$, $\dim(M') = \dim(M)+1$. Suppose the proposed move is such that $M'_e \neq M_e$, i.e., the effective networks of the current and the proposed networks are different. In our network-aware sampler, because we know the effective networks M'_e and M_e of the two models M' and M, respectively, we construct the following proposal:

$$\boldsymbol{f} := (\boldsymbol{k}^{\mathbf{1:}\boldsymbol{i}}_{M'_e}, \boldsymbol{k}^{\mathbf{1:}\boldsymbol{a}}_{M'_e}, \boldsymbol{w}^{\mathbf{1:}\boldsymbol{j}}_{M\setminus M_e}) = (\boldsymbol{k}^{\mathbf{1:}\boldsymbol{i}}_{M_e}, \boldsymbol{u}^{\mathbf{1:}\boldsymbol{a}}_{M_e}, \boldsymbol{k}^{\mathbf{1:}\boldsymbol{j}}_{M\setminus M_e}), \tag{4.12}$$

where $\boldsymbol{u}^{\mathbf{1:}\boldsymbol{a}}_{M} \sim \mathcal{N}(\boldsymbol{\mu}_M, \boldsymbol{\Sigma}_M)$ and $\boldsymbol{w}^{\mathbf{1:}\boldsymbol{j}} \sim p(\boldsymbol{k}^{\mathbf{1:}\boldsymbol{j}}_{M\setminus M_e}|M)$. The proposal mean $\boldsymbol{\mu}_M$:

$$\boldsymbol{\mu}_M = \underset{\boldsymbol{u}^{\mathbf{1:}\boldsymbol{a}}}{\arg\max}\; p(\boldsymbol{u}^{\mathbf{1:}\boldsymbol{a}}|\boldsymbol{k}^{\mathbf{1:}\boldsymbol{i}}_{M_e}, M'_e, \boldsymbol{\mathcal{D}}) \tag{4.13}$$

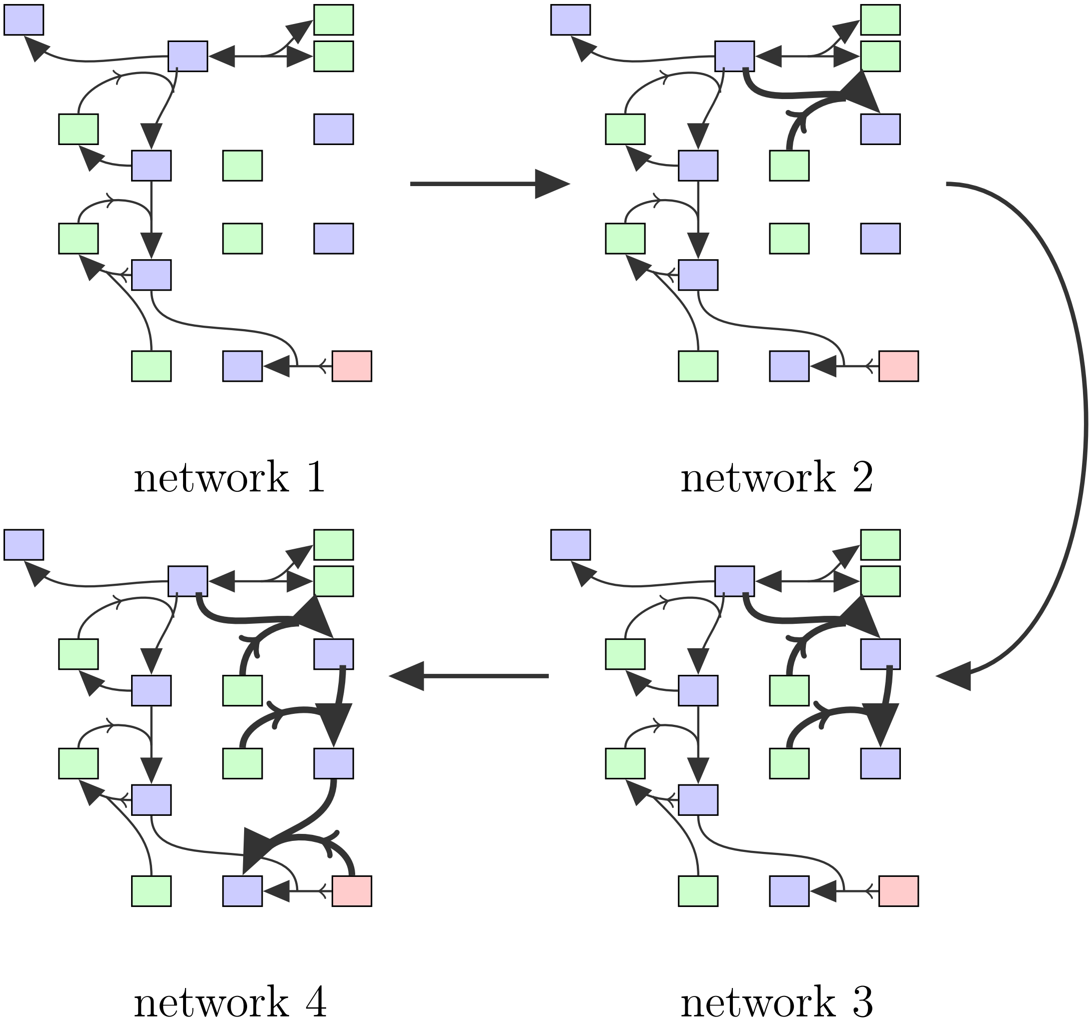


Figure 4-7: Model move from network 1 to 2 and 2 to 3 in the standard approach leads to the proposal adapting to the prior. Only the final move from 3 to 4 incorporates the likelihood function

Algorithm 7 Network-aware reversible jump MCMC

```
Given: A set of models $M \in \mathcal{M}$ with corresponding parameter vectors $\boldsymbol{k}_M$, posterior densities $p(M, \boldsymbol{k}_M|\boldsymbol{\mathcal{D}})$.
$\beta \in (0,1)$: probability of within-model move
Initlialize starting point $(M^0, \boldsymbol{k}_{M^0})$
for $n = 0$ to $N_{iter}$ do
    Sample $b \sim \mathcal{U}_{[0,1]}$
    if $b \leq \beta$ then
        Metropolis-Hastings within-model move
    else
        Sample $M' \sim q(M'|M^n = M)$; $M'_e$=eff$(M')$ and $M_e$=eff$(M)$
        if $|M'_e| > |M_e|$ then
  $$\boldsymbol{\mu}_M = \arg\max_{\boldsymbol{u}^{\mathbf{1:a}}} p(\boldsymbol{u}^{\mathbf{1:a}}|\boldsymbol{k}^{\mathbf{1:i}}_{M_e}, M'_e, \boldsymbol{\mathcal{D}}),\ \ \boldsymbol{\Sigma}_M = -\left[\nabla^2 \log p(\boldsymbol{u}^{\mathbf{1:a}}|\boldsymbol{k}^{\mathbf{1:i}}_{M_e}, M'_e, \boldsymbol{\mathcal{D}})\right]^{-1}\Big|_{\boldsymbol{\mu}_M}$$
            Sample $\boldsymbol{u}^{\mathbf{1:a}}_M \sim \mathcal{N}(\boldsymbol{\mu}_M, \boldsymbol{\Sigma}_M)$
            Set $(\boldsymbol{k}^{\mathbf{1:i}}_{M'_e}, \boldsymbol{k}^{\mathbf{1:a}}_{M'_e}, \boldsymbol{w}^{\mathbf{1:j}}_{M' \setminus M'_e}) = (\boldsymbol{k}^{\mathbf{1:i}}_{M_e}, \boldsymbol{u}^{\mathbf{1:a}}_M, \boldsymbol{k}^{\mathbf{1:j}}_{M' \setminus M'_e})$
            $\alpha((M, \boldsymbol{k}_M), (M', \boldsymbol{k'}_{M'})) = \min\left\{1, \frac{p(M'_e, \boldsymbol{k}_{M'_e}|\boldsymbol{\mathcal{D}}) q(M|M')}{p(M_e, \boldsymbol{k}_{M_e}|\boldsymbol{\mathcal{D}}) q(M'|M) \mathcal{N}(\boldsymbol{u}^{\mathbf{1:a}}_M; \boldsymbol{\mu}_M, \boldsymbol{\Sigma}_M)}\right\}$
        else if $|M'_e| < |M_e|$ then
  $$\boldsymbol{\mu}_{M'} = \arg\max_{\boldsymbol{u}^{\mathbf{1:a}}} p(\boldsymbol{u}^{\mathbf{1:a}}|\boldsymbol{k}^{\mathbf{1:i}}_{M_e}, M_e, \boldsymbol{\mathcal{D}}),\ \ \boldsymbol{\Sigma}_{M'} = -\left[\nabla^2 \log p(\boldsymbol{u}^{\mathbf{1:a}}|\boldsymbol{k}^{\mathbf{1:i}}_{M_e}, M_e, \boldsymbol{\mathcal{D}})\right]^{-1}\Big|_{\boldsymbol{\mu}_M}$$
            Sample $\boldsymbol{w}^{\mathbf{1:j}}_{M' \setminus M'_e} \sim p(\boldsymbol{k}^{\mathbf{1:j}}_{M' \setminus M'_e}|M')$
            Set $(\boldsymbol{k}^{\mathbf{1:i}}_{M'_e}, \boldsymbol{u}^{\mathbf{1:a}}_{M'}, \boldsymbol{k}^{\mathbf{1:j}}_{M' \setminus M'_e}) = (k^{\mathbf{1:i}}_{M_e}, \boldsymbol{k}^{\mathbf{1:a}}_{M_e}, \boldsymbol{w}^{\mathbf{1:j}}_{M' \setminus M'_e})$
            $\alpha((M, \boldsymbol{k}_M), (M', \boldsymbol{k'}_{M'})) = \min\left\{1, \frac{p(M'_e, \boldsymbol{k}_{M'_e}|\boldsymbol{\mathcal{D}}) q(M|M') \mathcal{N}(\boldsymbol{u}^{\mathbf{1:a}}_{M'}; \boldsymbol{\mu}_{M'}, \boldsymbol{\Sigma}_{M'})}{p(M_e, \boldsymbol{k}_{M_e}|\boldsymbol{\mathcal{D}}) q(M'|M)}\right\}$
        else
            Sample $\boldsymbol{w}^{\mathbf{1:j}}_{M' \setminus M'_e} \sim p(\boldsymbol{k}^{\mathbf{1:j}}_{M' \setminus M'_e}|M')$
            Set $(\boldsymbol{k}^{\mathbf{1:i}}_{M'_e}, \boldsymbol{u}^{\mathbf{1:a}}_{M \setminus M_e}, \boldsymbol{k}^{\mathbf{1:j}}_{M' \setminus M'_e}) = (\boldsymbol{k}^{\mathbf{1:i}}_{M_e}, \boldsymbol{k}^{\mathbf{1:a}}_{M \setminus M_e}, \boldsymbol{w}^{\mathbf{1:j}}_{M' \setminus M'_e})$
            $\alpha((M, \boldsymbol{k}_M), (M', \boldsymbol{k'}_{M'})) = 1$
        end if
        Sample $p \sim \mathcal{U}_{[0,1]}$
        if $p < \alpha((M, \boldsymbol{k}_M), (M', \boldsymbol{k'}_{M'}))$ then
            $(M^{n+1}, \boldsymbol{k}^{\boldsymbol{n+1}}_{M^{n+1}}) = (M', \boldsymbol{k}_{M'})$
        else
            $(M^{n+1}, \boldsymbol{k}^{\boldsymbol{n+1}}_{M^{n+1}}) = (M^n, \boldsymbol{k}^{\boldsymbol{n}}_{M^n})$
        end if
    end if
end for
```

is obtained by solving an a-dimensional optimization problem, where a is the difference between the number of reactions in the effective networks M_e' and M_e. The proposal covariance $\boldsymbol{\Sigma}_M$:

$$\boldsymbol{\Sigma}_M = -\left[\nabla\nabla \log p(\boldsymbol{u}^{\boldsymbol{1:a}}|\boldsymbol{k}^{\boldsymbol{1:i}}_{M_e}, M_e', \boldsymbol{\mathcal{D}})\right]^{-1}\Big|_{\boldsymbol{\mu}_M}, \tag{4.14}$$

is determined numerically using a finite-difference approximation at the proposal mean. Note $p(\boldsymbol{k}^{\boldsymbol{1:j}}_{M\setminus M_e}|M)$ is the prior probability density of reactions not in the effective network of M. The acceptance probability of the proposed move is given by

$$\alpha((M, \boldsymbol{k}_M), (M', \boldsymbol{k'}_{M'})) = \min\left\{1, A\right\}, \tag{4.15}$$

where

$$A = \frac{p(M_e', \boldsymbol{k}_{M_e'}|\boldsymbol{\mathcal{D}})q(M|M')}{p(M_e, \boldsymbol{k}_{M_e}|\boldsymbol{\mathcal{D}})q(M'|M)\mathcal{N}(\boldsymbol{u}^{\boldsymbol{1:a}}_M; \boldsymbol{\mu}_M, \boldsymbol{\Sigma}_M)}. \tag{4.16}$$

The reverse move has an acceptance probability $\min\{1, A^{-1}\}$. The idea behind the construction of our network aware proposals is that by solving for the conditional maximum of the joint posterior density of the reactions $M_e' \setminus M_e$ and determining the Hessian approximation at that point, we are building a Gaussian approximation of the conditional probability density $p(\boldsymbol{k_e'}|\boldsymbol{k_e}, M_e')$. In contrast, the standard network-unaware approach would not; in particular the second-order condition of Brooks et al. [20] produces a proposal that is the product of prior densities for $dim(M_e') - dim(M_e) - 1$ rate constants and the conditional posterior distribution of the final a^{th} rate constant.

Method	*Proposal*
Network unaware	$q_{nu}(\boldsymbol{k_e'} \setminus \boldsymbol{k_e}) \approx \prod_{i=1}^{a-1} p(k^i)p(\boldsymbol{k_e'} \setminus \boldsymbol{k^{1:a-1}}|\boldsymbol{k^{1:a-1}}, M_e', \boldsymbol{\mathcal{D}})$
Network aware	$q_{na}(\boldsymbol{k_e'} \setminus \boldsymbol{k_e}) \approx p(\boldsymbol{k_e'}|\boldsymbol{k_e}, M_e', \boldsymbol{\mathcal{D}})$

Table 4.1: Cluster switching parameter proposals

Mathematically, the two proposals are shown in Table 4.1. The steps of our network-aware reversible jump MCMC algorithm are given in Algorithm 7. If we think of the cluster $\{M_e, \boldsymbol{k_e}\}$ as the state space, the between-model moves that propose moves within the same cluster have an acceptance probability 1 with both the network-aware and the network-unaware approaches. However, when the proposed move is between two distinct clusters, our network aware approach chooses a proposal that approximates the joint conditional posterior density and hence leads to improved alignment between the densities $\tilde{p}(\boldsymbol{k_e}|\boldsymbol{\mathcal{D}})q(\boldsymbol{u}|\boldsymbol{k_e})$ and $\tilde{p}(\boldsymbol{k'_e}|\boldsymbol{\mathcal{D}})$ of the two spaces. Here $\tilde{p}(\boldsymbol{k_e}|\boldsymbol{\mathcal{D}})$ and $\tilde{p}(\boldsymbol{k_e}|\boldsymbol{\mathcal{D}})$ are the unnormalized densities of the rate constants of the two effective networks.

4.3.3 Sensitivity-based network-aware proposals

Between-model moves with deterministic reverse moves are a natural choice for nested models. However, in many cases, MCMC mixing may be improved by adopting non-deterministic reverse move types. In the context of reaction network inference, it is sometimes observed that a *maximum-a-posteriori* rate constant value for a reaction common to two networks differs substantially in the two networks. For example, consider the two networks in Figure 4-8. The most likely values for the rate constant of reaction * could differ significantly for the two networks. In such a case, keeping the rate constant of * fixed when proposing moves between the two networks leads to very poor acceptance rates. We propose a method to improve sampling efficiency of the network inference problem by identifying critical reactions common to the current and the proposed network and using a proposal distribution q for their rate constants in moves between the two networks. In other words, the reverse move from a high-dimensional effective network to a low-dimensional effective network is no longer deterministic. The move between the networks takes the following form:

$$(\boldsymbol{k}_{M'_e}, \boldsymbol{u'_e}) = \boldsymbol{f}(\boldsymbol{k}_{M_e}, \boldsymbol{u_e}). \tag{4.17}$$

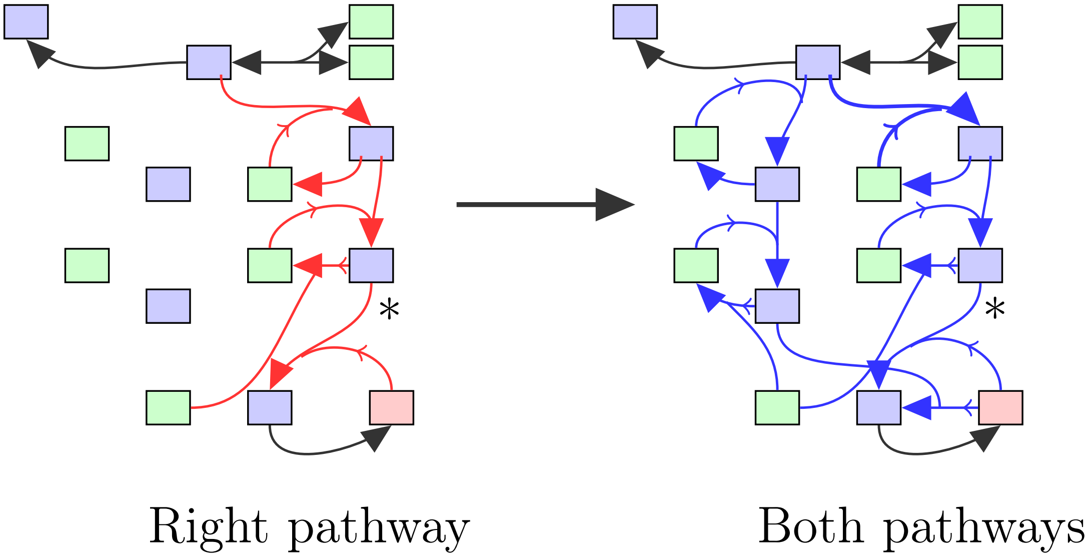


Figure 4-8: Two networks with different pathways

The question we answer next is how one could identify "key" reactions whose inclusion in the proposal would improve MCMC mixing at a limited computational overhead. Given a set of observables and the current and proposed network, a useful strategy is to identify the reactions to which the posterior density is most sensitive. To determine the sensitivity of the posterior density to individual reactions given a network, we employ local sensitivity analysis. Say, we have network M with reactions $R_1, R_2, ..., R_M$. We determine the expected local sensitivity index $\mathbb{E}\left[\left|\frac{\partial \log p(k_i|\boldsymbol{\mathcal{D}},\boldsymbol{k_{-i}})}{\partial k_i}\right|\right]$ of reaction i with $\boldsymbol{k}^*_{-i}$ given nominal values and the expectation taken with respect to the prior distribution $p(k_i|M)$. In practice, since the expectation is usually not analytically tractable, we settle for a noisy estimate of the expectation by evaluating the local sensitivity at a few realizations from the prior distribution and taking their average. Having determined the sensitivity of the log-posterior of the current and the proposed reaction network, we select a random number of high sensitivity reactions common to the two networks and include proposals for their rate constants in the forward and the reverse moves. The choice of the number of reactions to be included in the proposals is based on a Poisson distribution whose mean is kept at a small value. Choosing to include only a few common rate constant into the proposal is again based on the understanding that constructing effective proposals in high dimensions is generally hard. Thus, as the jump function for moves between models M and M' we have

$$\boldsymbol{f} := (\boldsymbol{k}_{M'}^{\mathbf{1}:\boldsymbol{i}}, \boldsymbol{k}_{M'}^{\mathbf{1}:\boldsymbol{a}}, \boldsymbol{k}_{M'}^{\mathbf{1}:\boldsymbol{c}}, \boldsymbol{u'}^{\mathbf{1}:\boldsymbol{c}}) = (\boldsymbol{k}_{M}^{\mathbf{1}:\boldsymbol{i}}, \boldsymbol{u}_{M}^{\mathbf{1}:\boldsymbol{a}}, \boldsymbol{u}^{\mathbf{1}:\boldsymbol{c}}, \boldsymbol{k}_{M}^{\mathbf{1}:\boldsymbol{c}}). \tag{4.18}$$

Here, $\{1 : i\}$ are indices of reactions whose parameter values are kept fixed during moves between models M and M', $\{1 : a\}$ are indices of reactions that are in model M' but not in M, and $\{1 : c\}$ are reactions that are present in both models but whose rate constant values are determined according to respective proposal distributions. Next, the parameter proposals $q(\boldsymbol{u}_{M}^{\mathbf{1}:\boldsymbol{a}}, \boldsymbol{u}^{\mathbf{1}:\boldsymbol{c}}|\boldsymbol{k}_{M}^{\mathbf{1}:\boldsymbol{i}})$ and $q(\boldsymbol{u'}^{\mathbf{1}:\boldsymbol{c}}|\boldsymbol{k}_{M'}^{\mathbf{1}:\boldsymbol{i}})$ are again chosen as Gaussian approximations of the conditional posteriors $p(\boldsymbol{k}_{M'}^{\mathbf{1}:\boldsymbol{a}}, \boldsymbol{k}_{M'}^{\mathbf{1}:\boldsymbol{c}}|\boldsymbol{k}_{M'}^{\mathbf{1}:\boldsymbol{i}}, \boldsymbol{\mathcal{D}})$ and $p(\boldsymbol{k}_{M}^{\mathbf{1}:\boldsymbol{c}}|\boldsymbol{k}_{M}^{\mathbf{1}:\boldsymbol{i}}, \boldsymbol{\mathcal{D}})$, respectively. Note, this construction of parameter proposals improves alignment between densities $\tilde{p}(\boldsymbol{k}_{M'}^{\mathbf{1}:\boldsymbol{i}}, \boldsymbol{k}_{M'}^{\mathbf{1}:\boldsymbol{a}}, \boldsymbol{k}_{M'}^{\mathbf{1}:\boldsymbol{c}}|M', \boldsymbol{\mathcal{D}})q(\boldsymbol{u'}^{\mathbf{1}:\boldsymbol{c}})$ and $\tilde{p}(\boldsymbol{k}_{M}^{\mathbf{1}:\boldsymbol{i}}, \boldsymbol{k}_{M}^{\mathbf{1}:\boldsymbol{c}}|M, \boldsymbol{\mathcal{D}})q$ $(\boldsymbol{u}_{M}^{\mathbf{1}:\boldsymbol{a}}, \boldsymbol{u}^{\mathbf{1}:\boldsymbol{c}})$ and produces efficient reversible jump proposals. The above construction of reversible jump proposals satisfies the second-order conditions of Brooks et al [20]. The foregoing discussion has focused on a network-unaware approach, where the effective networks are not known apriori. As we discussed in Section 4.3.2 on network-aware proposals, for the network inference problem, improved proposals that approximate the joint posterior conditionals of the rate constants of the difference in reactions between the current and proposed effective networks can be constructed by determining the effective networks of proposed networks. We combine the network-aware scheme of the previous section with the sensitivity-based determination of move types to yield the the *sensitivity-based network-aware proposals*. The sequence of steps for our sensitivity-based network-aware reversible jump MCMC algorithm are given in Algorithm 8.

Algorithm 8 Sensitivity-based network-aware reversible jump MCMC

```
Given: A set of models $M \in \mathcal{M}$ with corresponding parameter vectors $\boldsymbol{k}_M$, posterior densities $p(M, \boldsymbol{k}_M|\boldsymbol{\mathcal{D}})$.
$\beta \in (0, 1)$: probability of within-model move
Initlialize starting point $(M^0, \boldsymbol{k}_{M^0})$
for $n = 0$ to $N_{iter}$ do
    Sample $b \sim \mathcal{U}_{[0,1]}$
    if $b \leq \beta$ then
        Metropolis-Hastings within-model move
    else
        Sample $M' \sim q(M'|M^n = M)$; $M'_e$=eff$(M')$ and $M_e$=eff$(M)$
        $r_1 \sim Poisson(1.5)$ and $r_2 \sim Poisson(1.5)$
        $\{1 : c\}$=reactions with top $r_1$ and $r_2$ sensitivities of $M_e$ and $M'_e$, respectively, and common to $M_e$ and $M'_e$.
        if $|M'_e| > |M_e|$ then
            $\boldsymbol{\mu}_{M_e} = \underset{\boldsymbol{u^{1:a}}, \boldsymbol{u^{1:c}}}{\arg\max}\ p(\boldsymbol{u^{1:a}}, \boldsymbol{u^{1:c}}|\boldsymbol{k}^{\boldsymbol{1:i}}_{M_e}, M'_e, \boldsymbol{\mathcal{D}}),\ \boldsymbol{\Sigma}_{M_e} = -\left[\nabla^2 \log p(\boldsymbol{u^{1:a}}, \boldsymbol{u^{1:c}}|\boldsymbol{k}^{\boldsymbol{1:i}}_{M_e}, M'_e, \boldsymbol{\mathcal{D}})\right]^{-1}|_{\boldsymbol{\mu}_M}$
            $\boldsymbol{\mu}_{M'_e} = \underset{\boldsymbol{u^{1:c}}}{\arg\max}\ p(\boldsymbol{u^{1:c}}|\boldsymbol{k}^{\boldsymbol{1:i}}_{M_e}, M_e, \boldsymbol{\mathcal{D}}),\ \boldsymbol{\Sigma}_{M'_e} = -\left[\nabla^2 \log p(\boldsymbol{u^{1:c}}|\boldsymbol{k}^{\boldsymbol{1:i}}_{M_e}, M_e, \boldsymbol{\mathcal{D}})\right]^{-1}|_{\boldsymbol{\mu}_{M'}}$
            Sample $\boldsymbol{u^{1:a}}, \boldsymbol{u^{1:c}} \sim \mathcal{N}(\boldsymbol{\mu}_{M_e}, \boldsymbol{\Sigma}_{M_e})$
            Set $(\boldsymbol{k}^{\boldsymbol{1:i}}_{M'_e}, \boldsymbol{k}^{\boldsymbol{1:a}}_{M'_e}, \boldsymbol{k}^{\boldsymbol{1:c}}_{M'_e}, \boldsymbol{u'^{1:c}}) = (\boldsymbol{k}^{\boldsymbol{1:i}}_{M_e}, \boldsymbol{u^{1:a}}, \boldsymbol{u^{1:c}}, \boldsymbol{k}^{\boldsymbol{1:c}}_{M_e})$
        else if $|M'_e| < |M_e|$ then
            $\boldsymbol{\mu}_{M_e} = \underset{\boldsymbol{u^{1:c}}}{\arg\max}\ p(\boldsymbol{u^{1:c}}|\boldsymbol{k}^{\boldsymbol{1:i}}_{M_e}, M'_e, \boldsymbol{\mathcal{D}}),\ \boldsymbol{\Sigma}_{M_e} = -\left[\nabla^2 \log p(\boldsymbol{u^{1:c}}|\boldsymbol{k}^{\boldsymbol{1:i}}_{M_e}, M'_e, \boldsymbol{\mathcal{D}})\right]^{-1}|_{\boldsymbol{\mu}_M}$
            $\boldsymbol{\mu}_{M'_e} = \underset{\boldsymbol{u^{1:a}}, \boldsymbol{u^{1:c}}}{\arg\max}\ p(\boldsymbol{u^{1:a}}, \boldsymbol{u^{1:c}}|\boldsymbol{k}^{\boldsymbol{1:i}}_{M_e}, M_e, \boldsymbol{\mathcal{D}}),\ \boldsymbol{\Sigma}_{M'_e} = -\left[\nabla^2 \log p(\boldsymbol{u^{1:a}}, \boldsymbol{u^{1:c}}|\boldsymbol{k}^{\boldsymbol{1:i}}_{M_e}, M_e, \boldsymbol{\mathcal{D}})\right]^{-1}|_{\boldsymbol{\mu}_{M'}}$
            Sample $\boldsymbol{u^{1:c}} \sim \mathcal{N}(\boldsymbol{\mu}_{M_e}, \boldsymbol{\Sigma}_{M_e})$
            Set $(\boldsymbol{k}^{\boldsymbol{1:i}}_{M'_e}, \boldsymbol{u'^{1:a}}, \boldsymbol{u'^{1:c}}, \boldsymbol{k}^{\boldsymbol{1:c}}_{M'_e}) = (\boldsymbol{k}^{\boldsymbol{1:i}}_{M_e}, \boldsymbol{k}^{\boldsymbol{1:a}}_{M_e}, \boldsymbol{k}^{\boldsymbol{1:c}}_{M_e}, \boldsymbol{u^{1:c}})$
        else $\boldsymbol{\mu}_{M_e} = \emptyset$, $\boldsymbol{\mu}_{M'_e} = \emptyset$, $\boldsymbol{\Sigma}_{M_e} = \emptyset$, $\boldsymbol{\Sigma}_{M'_e} = \emptyset$
        end if
        Sample $\boldsymbol{k}^{\boldsymbol{1:j}}_{M' \setminus M'_e} \sim p(\boldsymbol{k}^{\boldsymbol{1:j}}_{M' \setminus M'_e}|M')$
        Sample $p \sim \mathcal{U}_{[0,1]}$
        if $p < \min\left\{1, \frac{p(M'_e, \boldsymbol{k}_{M'_e}|\boldsymbol{\mathcal{D}})q(M^n|M')\mathcal{N}(\boldsymbol{u'}; \boldsymbol{\mu}_{M'_e}, \boldsymbol{\Sigma}_{M'_e})}{p(M_e, \boldsymbol{k}_{M_e}|\boldsymbol{\mathcal{D}})q(M'|M^n)\mathcal{N}(\boldsymbol{u}; \boldsymbol{\mu}_{M_e}, \boldsymbol{\Sigma}_{M_e})}\right\}$ then
            $(M^{n+1}, \boldsymbol{k}^{\boldsymbol{n+1}}_{M^{n+1}}) = (M', \boldsymbol{k}_{M'})$
        else
            $(M^{n+1}, \boldsymbol{k}^{\boldsymbol{n+1}}_{M^{n+1}}) = (M^n, \boldsymbol{k}^{\boldsymbol{n}}_{M^n})$
        end if
    end if
end for
```

4.3.4 Derandomization of conditional expectations

The above Algorithms 7 and 8 lead to gains in sampling efficiency compared to a reversible jump MCMC algorithm that does not use information on network structure in designing between-model moves and parameter proposals. Identifying clusters of models can be further used for additional variance reduction. With the knowledge that all models belonging to the same cluster have identical model evidence, we can compute some expectations analytically and thereby obtain posterior averages of features with lower variances.

General formulation

Let us assume we performing model inference with F as one the quantities of interest. Generally, we may be interested in quantities such as the posterior model probabilities, reaction inclusion probabilities of reactions, or pathway probabilities. The Monte Carlo estimate of F from posterior samples can be written as:

$$\begin{aligned}
\hat{F} &= p(F=1|\boldsymbol{\mathcal{D}}) \\
&= \int p(F=1|C)p(C|\boldsymbol{\mathcal{D}})dC \\
&= \int p(F=1|M)p(M|C)p(C|\boldsymbol{\mathcal{D}})dMdC \\
&= \int \mathbb{E}_{p(M|C)}\left[p(F=1|M)\right]p(C|\boldsymbol{\mathcal{D}})dC \\
&= \frac{1}{N_s}\sum_{i=1}^{N_s}\mathbb{E}_{p(M|C^i)}\left[p(F=1|M)\right],
\end{aligned}$$

where C refers to model clusters, N_s is the number of posterior samples and $\boldsymbol{\mathcal{D}}$ the available data. In the above equation, $\mathbb{E}_{p(M|C^i)}[p(F=1|M)]$ is the expected value of $p(F=1|M)$ conditioned on the generated sample C^i. Knowing the cluster to which each sample belongs and the dependence of the feature on the models included in the cluster, the above expectation can be computed analytically and allows variance reduction. In contrast, in the network-unaware approach, the expectation is computed through Monte Carlo sampling.

Example: model probability estimates

Consider that the feature of interest is the probability of model m. Thus, applying the above formula to the estimation of model probability, we get

$$\begin{aligned}
\hat{M}_m &= p(M_m = 1|\boldsymbol{\mathcal{D}}) \\
&= \int p(M_m = 1|C)p(C|\boldsymbol{\mathcal{D}})dC \\
&= \int p(M_m = 1|M)p(M|C)p(C|\boldsymbol{\mathcal{D}})dMdC \\
&= \int \mathbb{E}_{p(M|C)}[p(M_m = 1|M)]p(C|\boldsymbol{\mathcal{D}})dC \\
&= \frac{1}{N_s}\sum_{i=1}^{N_s} \mathbb{E}_{p(M|C^i)}\left[p(M_m = 1|M)\right] \\
&= \frac{1}{N_s}\sum_{i=1}^{N_s} p(M_m|C_K)\mathbb{1}_{C_K}(C^i), \qquad (4.19)
\end{aligned}$$

where $K : \mathbb{1}_{M_m \in C_K}(M_m) = 1$ and $\mathbb{1}$ is the indicator function. In our network aware schemes, $p(M_m|C_K)$ can be computed analytically. For example, for a cluster C_K with N_K models, taking the prior distribution over models to be uniform, the model probability estimate is

$$\hat{M}_m = \frac{1}{N_s}\sum_{i=1}^{N_s} \frac{1}{N_K}\mathbb{1}_{C_K}(C^i) \qquad (4.20)$$

In contrast, with a standard reversible-jump algorithm, the model probability estimate is

$$\hat{M}_m = \frac{1}{N_s}\sum_{i=1}^{N_s} \mathbb{1}_{M_m}(M^i)\mathbb{1}_{C_K}(C^i) \tag{4.21}$$

4.4 Results

We present four example problems and demonstrate the efficiency of our network-aware sampling approaches compared to the network-unaware second-order approach of Brooks et al [20]. The observables in our examples are species concentrations and the concentration evolution is modeled using the law of mass action/Michaelis-Menten functionals. The resulting nonlinear system of ordinary differential equations are solved using the multistep BDF integrator available in the SUNDIALS suite [69].

4.4.1 Setup of the Bayesian model inference problem

Before discussing the four model inference examples individually, we describe the choices we make for the likelihood function and prior distribution in our Bayesian formulation. In the following, we use $\tilde{k}$ to refer to the rate constants of the reactions that are treated as uncertain and $\hat{k}$ to denote the rate constants of reactions that are kept fixed. By "fixed," we mean that a particular reaction is always included in the model and that its rate constant is not a target of the inference procedure.

Likelihood function

As described in Section 3.1, evaluating the posterior probability in the Bayesian approach requires evaluating the likelihood function $p(\boldsymbol{\mathcal{D}}|\boldsymbol{k})$, where $\boldsymbol{\mathcal{D}}$ are the data and $\boldsymbol{k} = (\tilde{k}, \hat{k})$ are the reaction parameters. We employ an i.i.d. additive Gaussian model for the difference between model predictions and observations; thus the data are represented as

$$\boldsymbol{\mathcal{D}} = \boldsymbol{G}(\tilde{k}, \hat{k}) + \boldsymbol{\epsilon_n}, \tag{4.22}$$

where $\boldsymbol{\epsilon_n} \sim \mathcal{N}_n(\mathbf{0}, \sigma^2 \boldsymbol{I_n})$, n is the number of observations, $\boldsymbol{I_n}$ is an n-by-n identity matrix, and $\boldsymbol{G}(\tilde{k}, \hat{k})$ is the prediction of the forward model at the given value of the reaction parameters. The specific values of the noise standard deviations σ are given later. The deterministic predictions $\boldsymbol{G}(\tilde{k}, \hat{k})$ are obtained with the ODE integrator. The likelihood function is thus given by

$$\begin{aligned} p(\boldsymbol{\mathcal{D}}|\bar{k}) &= \mathcal{N}_n(\boldsymbol{\mathcal{D}}|\boldsymbol{G}(\tilde{k}, \hat{k}), \sigma^2 \boldsymbol{I_n}) \\ &= \prod_{t=1}^{n} \mathcal{N}(\boldsymbol{\mathcal{D}}|\boldsymbol{G}(\tilde{k}, \hat{k}), \sigma^2) \\ &= \prod_{t=1}^{n} \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(d^t - \boldsymbol{G}(\tilde{k}, \hat{k}))^2}{2\sigma^2}\right), \end{aligned} \tag{4.23}$$

where d^t are components of the data vector $\boldsymbol{\mathcal{D}}$.

Prior specification

Since reaction rate constants must be positive, while their uncertainties may span multiple orders of magnitude, we take the prior distribution to be an independent log-normal distribution on each rate constant. That is,

$$p(k_i) : \log_{10} k_i \sim \mathcal{N}(\mu_{p,i}, \sigma_{p,i}^2). \tag{4.24}$$

One could even encode prior correlations among the rate constants. The model prior distributions $p(M)$ in the following examples are uniform unless explicitly mentioned otherwise.

4.4.2 Example 1: linear Gaussian network inference

In our first example, we consider a six-dimensional reaction network (Figure 4-9) in which the species interactions are modeled as linear Gaussian functions, i.e.,

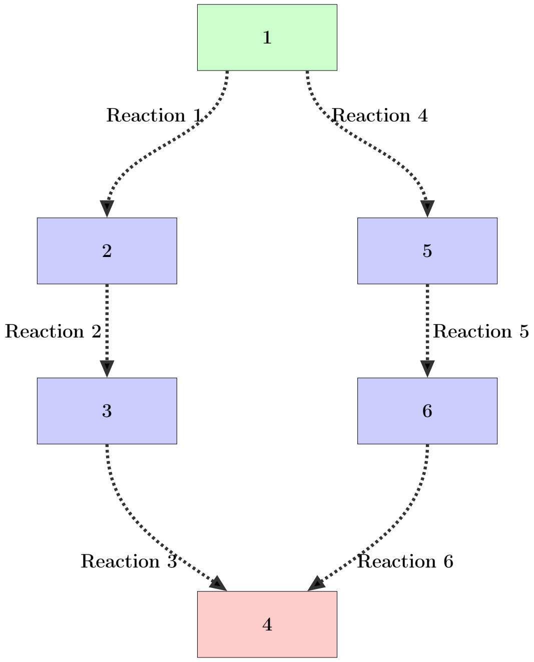


Figure 4-9: 6 uncertain reactions; species 1 has non-zero initial concentration, species 2, 3, 5, and 6 are produced in operation, and species 4 is observed.

$$c_i = \sum_j a_j c_j + \epsilon_i. \tag{4.25}$$

Here, c_i is the concentration of species i and c_j is the concentration of species j that i directly depends on. a_j are the unknown rate constants. We take species ④ as the only observable and generate 10 i.i.d. data points with $\{a_i\} = \{0, 0, 0, 2, 1, 2\}$ and noise model $\epsilon_4 = \mathcal{N}(0, 2)$. All other variances ϵ_i are identically set to zero. Taking a Gaussian prior distribution on $\{a_j\}$ and a Gaussian noise model ϵ_i yields a posterior distribution on $\{a_j\}$ that is a multivariate Gaussian. In addition, the marginal likelihood of data $\mathcal{D}$ in this case is available in closed form. Specifically, for the present example we impose independent Gaussian priors on a_is with a mean vector $\boldsymbol{\mu}_p = (0, 0, 0, 0, 0, 0, 0)$ and variance vector $\boldsymbol{\sigma}_p^2 = (10, 10, 10, 10, 10, 10)$. The noise variance for the likelihood function, as in the data generating process, is taken

Method[†]	$p(M)^a$	$\bar{\alpha}_M^b$	$\bar{\alpha}_C^c$	ESS^d	ESS/min
Network unaware	0.401	0.450	0.121	57	44.4
Network aware	0.402	0.479	0.169	482	210.0

†: Performance is averaged over 10 replications
a: Posterior probability of the data-generating model
b: Between-model acceptance rate
c: Between-cluster acceptance rate
d: Effective sample size for 10000 samples

Table 4.2: Summary statistics of MCMC simulations (Example 1).

as $\sigma^2 = 2$.

We compare the sampling efficiency of our proposed network-aware algorithm (Section 4.3.2) to the network-unaware second-order method of Brooks et al [20]. We simulated 10 MCMC chains of 300000 samples using both the approaches. All simulations produce similar posterior inferences, thereby indicating MCMC convergence. Table 4.2 shows the acceptance rates of between model and between-cluster moves for the two schemes. A high model-switching acceptance rate with the same posterior inference is usually regarded as an indication of superior mixing. We find that the acceptance rates are higher with our network-aware proposals. Effective sample size (ESS) calculation for statistics that retain interpretation throughout the simulation is another diagnostic for MCMC mixing. Effective sample size gives the equivalent number of independent samples to the dependent MCMC samples obtained in terms of learning the particular statistic. We take the number of reactions in the models as the quantity whose ESS is compared. Again in Table 4.2 we see that our network-aware scheme has an ESS that is roughly nine times the ESS obtained using the network-unaware approach and hence provides a more efficient posterior simulation. A more useful comparison of the two schemes would be to compare the number of effective samples per unit time because it also incorporates the computational time. In last column of Table 4.2, we give the ESS per minute and again find our network-aware is also computationally efficient.

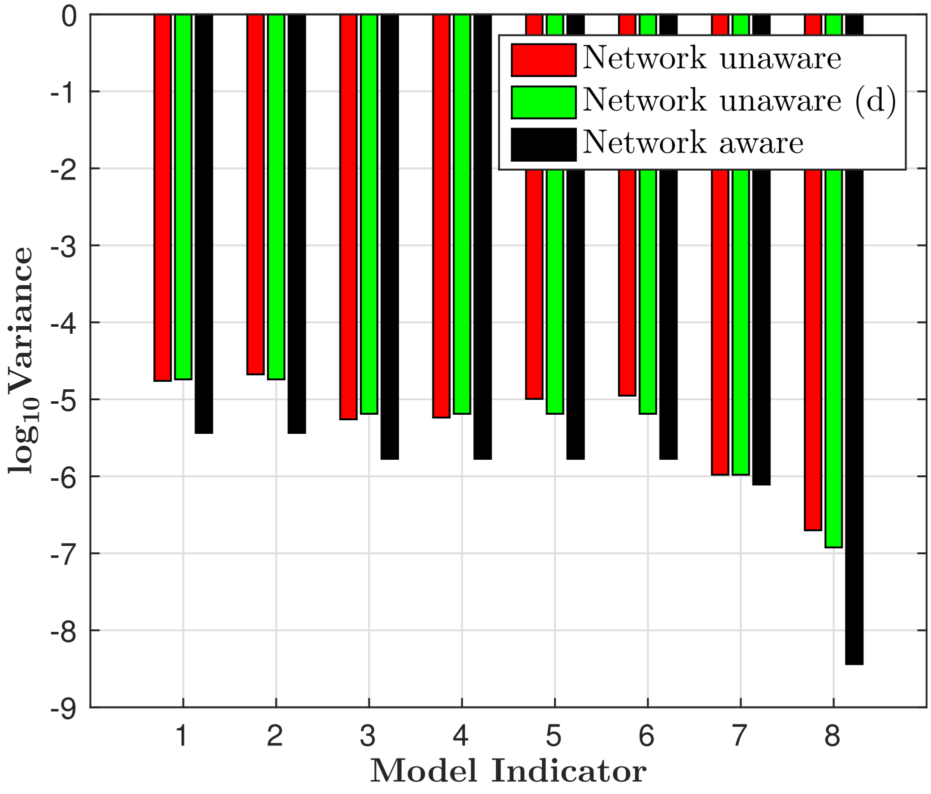


Figure 4-10: Variance of the eight highest posterior model probability estimates in Example 1

Further in Figure 4-10, we present the variance of the estimated posterior model probabilities. The variance estimates are calculated based on 10 chains. The results of the network-aware sampler are compared to the network-unaware approach, with and without derandomization. Derandomization in the network-unaware approach refers to performing the MCMC sampling without acknowledging the network structure, but using that knowledge only as a post processing variance reduction method. We see clear benefit in using out network-aware sampling approach compared to the standard method with a 3–4 times reduction in variance.

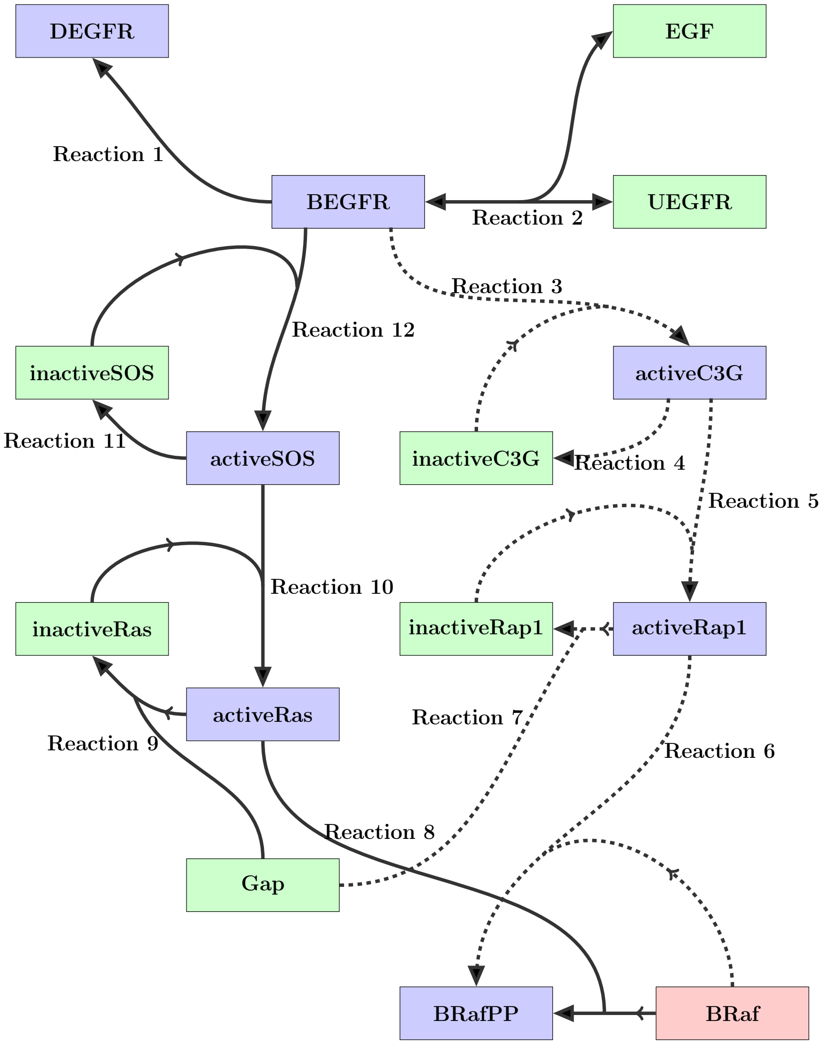


Figure 4-11: A reaction network with 5 (reactions 3, 4, 5, 6, and 7) uncertain reactions. BRaf is the observable.

4.4.3 Example 2: five-dimensional nonlinear network inference

As our second example, we consider a five-dimensional nonlinear network inference problem where the species interactions are governed by the law of mass action (Figure 4-11). The law of mass action gives the rate of a chemical reaction (say $X + Y \rightarrow Z$) as the product of a reaction-specific rate constant k with reactant concentrations $[X]$ and $[Y]$:

$$\text{Rate} = -k[X][Y]. \tag{4.26}$$

Under some assumptions, the law of mass action produces Michaelis-Menten reaction rate expression

$$Rate = \frac{k[S]}{k_M + [S]}, \tag{4.27}$$

or when enzyme concentration is taken into account [91]:

$$Rate = k[E]_0 \frac{[S]}{k_M + [S]}, \tag{4.28}$$

where k denotes the rate constant, $[E]_0$ is the enzyme concetration, $[S]$ the substrate concentration, and k_M the Michaelis constant.

In the present example, we consider a subset of reactions (15 species and 12 reactions) proposed for a protein-signalling network of the activation of extracellular signal-regulated kinase (ERK) by epidermal growth factor (EGF) (Figure 4-11) [118]. The ODE forward model governing the evolution of species concentrations is described in detail in Appendix B.1. We keep reactions 1, 2, 8, 9, 10, 11, and 12 fixed (denoted by thick lines in the reaction graph 4-11 and shaded pink in Table 4.3) and thus they are included in all the inferred models. The rate constants of all fixed reactions and Michaelis constants of all reactions are set to their base values (Table 4.3). Reactions

3, 4, 5, 6, and 7 are taken to be uncertain and the concentration of BRaf is taken to be the observable. With the above five uncertain reactions, the number of potential models is 32. And with only BRaf as the observable, the number of clusters is 5. The resulting problem is a nonlinear network inference problem for which the marginal likelihood is not analytically computable.

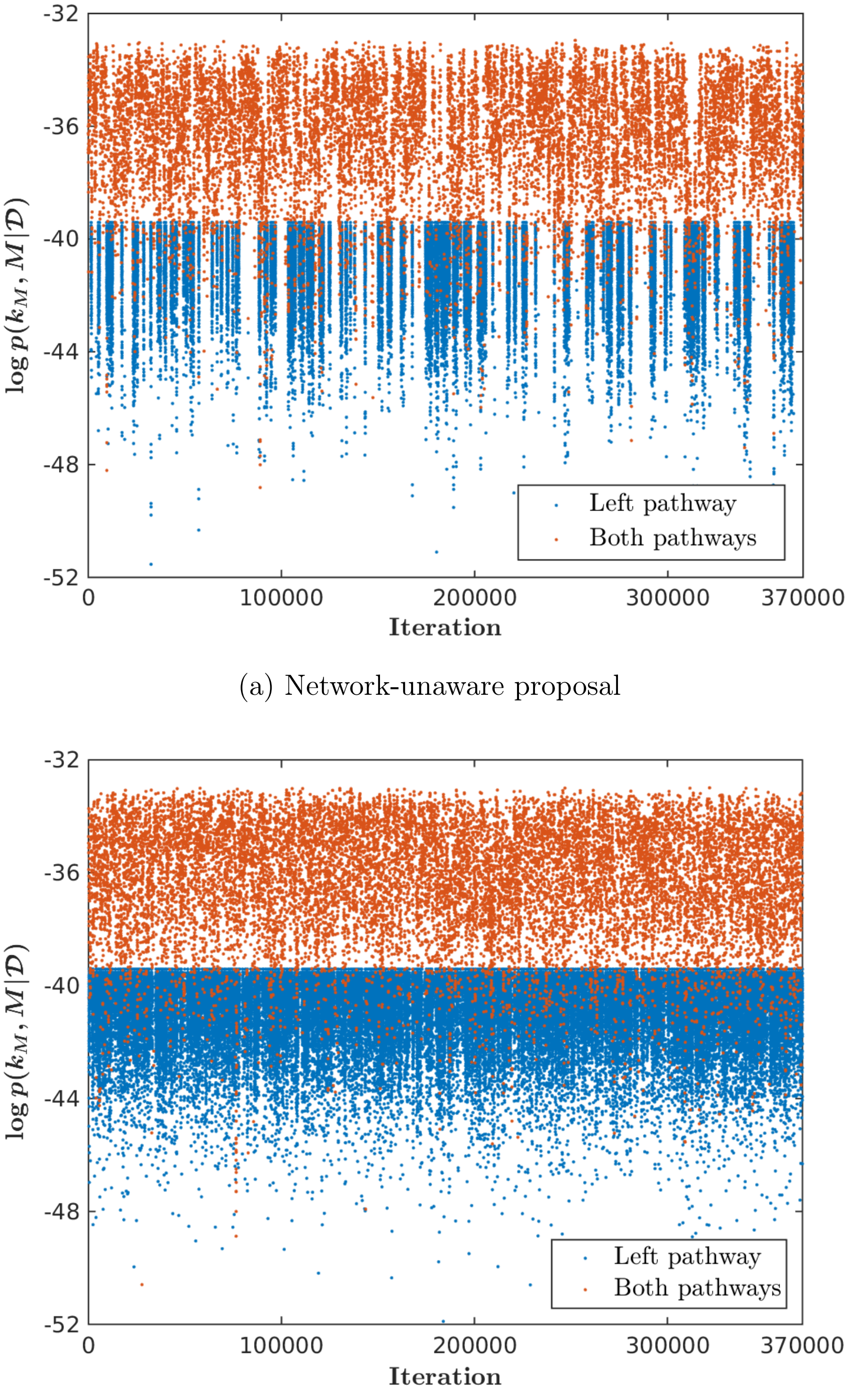
	Reaction	$\log_{10} k^{*a}$	k_M^b	Prior uncertainty
1	BEGFR → DEGFR	0.0	-	–
2a	EGF + UEGFR → BEGFR	1.5	-	–
2b	BEGFR → EGF + UEGFR	0.0	-	–
3	inactiveC3G+BEGFR → activeC3G+BEGFR	0.5	3386.3875	$\log_{10} k = \mathcal{N}(1.1, 0.2)$
4	activeC3G → inactiveC3G	2.0	-	$\log_{10} k = \mathcal{N}(1.4, 0.2)$
5	inactiveRap1+activeC3G → activeRap1+activeC3G	2.0	3566	$\log_{10} k = \mathcal{N}(2.6, 0.2)$
6	BRaf+activeRap1 → BRafPP+activeRap1	0.4	17991.179	$\log_{10} k = \mathcal{N}(1.0, 0.2)$
7	activeRap1+Gap → inactiveRap1+Gap	1.0	6808.32	$\log_{10} k = \mathcal{N}(0.4, 0.2)$
8	BRaf+activeRas → BRafPP+activeRas	0.5	7631.63	–
9	activeRas+Gap → inactiveRas+Gap	0.0	12457.816	–
10	inactiveRas+activeSOS → activeRas+activeSOS	0.5	13.73	–
11	activeSOS → inactiveSOS	4.0	9834.13	–
12	inactiveSOS+BEGFR → activeSOS+BEGFR	2.5	8176.56	–

[a] logarithm (base rate constant value)
[b] Base value of Michaelis constant (Obtained from Xu et al. [118])

Table 4.3: Proposed reactions for Example 2

We generated 20 i.i.d. data points with noise model $\mathcal{N}(0, 4)$ and rate constants and Michaelis constants set to their base values (Table 4.3). We impose independent Gaussian priors on the rate constants of the uncertain reactions with means and variances as shown in Table 4.3. The above prior amounts to roughly three orders of magnitude prior uncertainty in the rate constants. All models are assigned a uniform prior probability. The noise variance for the likelihood function, as in the data generating process, is taken as $\sigma^2 = 4$.

We compare the sampling efficiency of our network-aware algorithm (Section 4.3.2) to the network-unaware 2^{nd} order proposal of Brooks et al [20]. We simulated 5 replications of 400000 samples using both the approaches. 30000 samples each were discarded as burn-in. All simulations produce similar posterior inferences, thereby indicating convergence. Figure 4-12 shows samples generated from the posterior distribution using the two approaches color coded according to the pathway they belong.



(a) Network-unaware proposal

(b) Network-aware proposal

Figure 4-12: MCMC trace plots for Example 2: posterior samples from models with both pathways in orange and samples from models with only the left pathway in blue

Blue points are posterior samples from all models that belong to the left pathway, i.e., models that do not contain any of reactions 3, 5, and 6. Orange points are posterior samples from models that operate with both—left and right branch—pathways, i.e., models that necessarily include reactions 3, 5, 6, 8, 10 and 12. The higher frequency of moves between the left pathway models and both pathways models with the network-aware approach is a sign of faster posterior exploration and consequently better MCMC mixing. Table 4.4 shows the acceptance rates of between models moves and between-cluster moves for the two approaches. High model-switching and cluster-switching acceptance rates with the same posterior inference is an indication of superior mixing of the network-aware approach. In Table 4.4, we also present the ESS for the number-of-reactions-in-model statistic. The network-aware scheme has a ten-fold higher ESS compared to the network-unaware approach. The ESS per minute diagnostic in the last column of Table 4.4 confirms favourability of the network-aware approach with the computational cost taken into account. The absolute value of ESS per minute depends on the relative and absolute tolerance settings of the ODE solver. In particular, we chose very tight tolerances, but higher ESS/min can be obtained with loose tolerances. Nonetheless, the relative values of ESS/min demonstrate the advantage of using the network-aware sampling approach.

Method[†]	$p(M)^a$	$\bar{\alpha}_M^b$	$\bar{\alpha}_C^c$	ESS[d]	ESS/min[e]
Network unaware	0.7545	0.19	0.015	10	0.175
Network aware	0.7544	0.22	0.034	110	0.301

†: Performance is averaged over 5 simulation runs
a: Posterior probability of the data-generating model
b: Between-model move acceptance rate
c: Between-cluster move acceptance rate
d: Effective sample size for 10000 samples
e: The absolute values depend on the tolerances chosen for the ODE solver

Table 4.4: Summary statistics of MCMC simulations (Example 2).

Finally, in Figure 4-13, we present the variance of the estimated posterior model

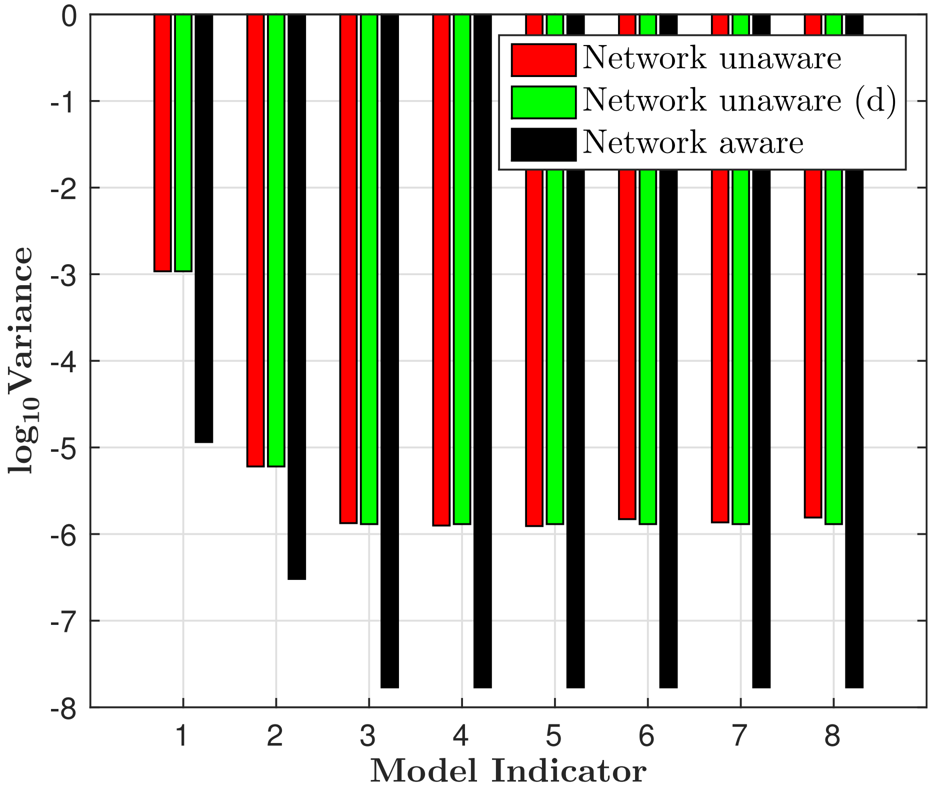


Figure 4-13: Variance of the eight highest posterior model probability estimates in Example 2

probabilities. The variance estimates are calculated based on 5 independent chains. The results of the network-aware sampler are compared to the network-unaware approach, with and without derandomization. We can see two-orders of magnitude lower variance values using our network-aware sampling approach, further supporting the merit in adopting the network-aware approach.

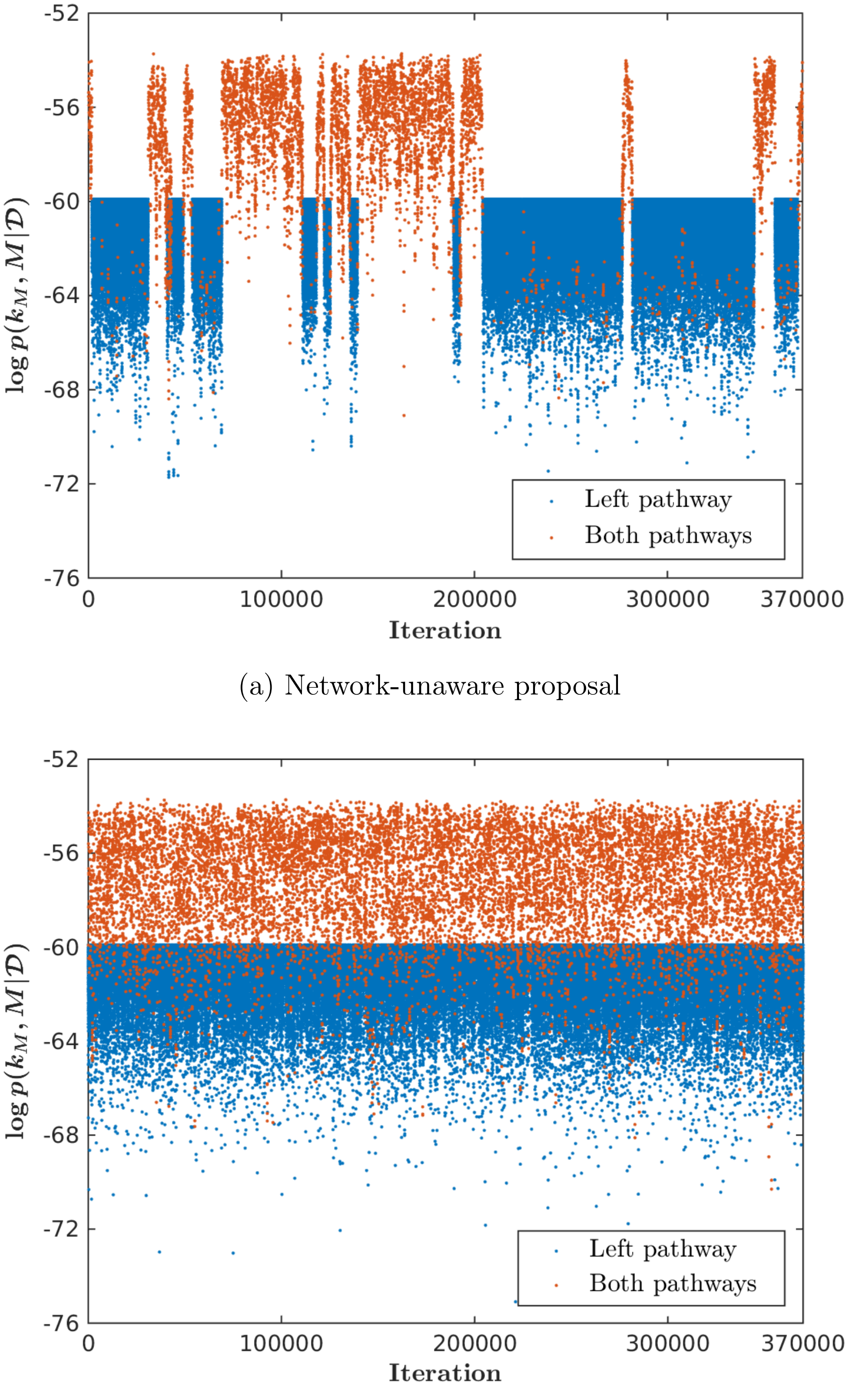
We compared the network-aware and network-unaware approaches on another problem for which the posterior samples are shown in Figure 4-14. The setup of this problem is identical to the Example just presented, except with a different prior on the rate constants. Again the blue points correspond to posterior samples from models belonging to left pathway and orange points correspond to samples from models that has both—left and right—pathways. Visually, we can see that mixing with the network-unaware approach is poor, whereas with the network-aware approach continues to be good.

4.4.4 Example 3: six-dimensional nonlinear network inference

In our third example, we consider a six-dimensional nonlinear network inference problem where species interactions are governed by law of mass action (Figure 4-15). Once again, we consider a protein-signalling network consisting of 15 species and 12 potential species interactions (Figure 4-15). The ODE forward model governing the evolution of species concentrations is described in detail in Appendix B.1. We keep reactions 1, 2, 4, 7, 9, and 11 fixed (denoted by thick lines in the reaction graph 4-15 and shaded pink in Table 4.5), and thus they are included in all the inferred models. The rate constants of all fixed reactions and Michaelis constants of all reactions are set to their base values. Reactions 3, 5, 6, 8, 10, and 12 are uncertain and the concentration of BRaf is again the observable. With the above six uncertain reactions, the number of potential models is 64. However, with only BRaf as the observable, the number of clusters is 4.

We generated 20 i.i.d. data points with noise model $\mathcal{N}(0, 0.25)$ and all rate constants and Michaelis constants set to their base values (Table 4.5). We impose independent Gaussian priors on the rate constants of the uncertain reactions with means and variances as shown in Table 4.5. The above prior amounts to roughly two orders of magnitude prior uncertainty in the rate constants. The prior probability distribution over models is set such that all models except one are given uniform probability and the model with all uncertain reactions included a prior probability five times the other models. This choice is made to ensure that model with all reactions included (thus with both pathways active) does not have insignificant posterior probability. The noise variance for the likelihood function, as in the data generating process, is taken as $\sigma^2 = 0.25$.

Samplers that did not use sensitivity-based proposals show very poor mixing. Figure 4-16 shows samples generated (after a burn-in of 30000 samples) from the pos-



(a) Network-unaware proposal

(b) Network-aware proposal

Figure 4-14: MCMC trace plots for Example 2-additional: posterior samples from models with both pathways in orange and samples from models with only the left pathway in blue

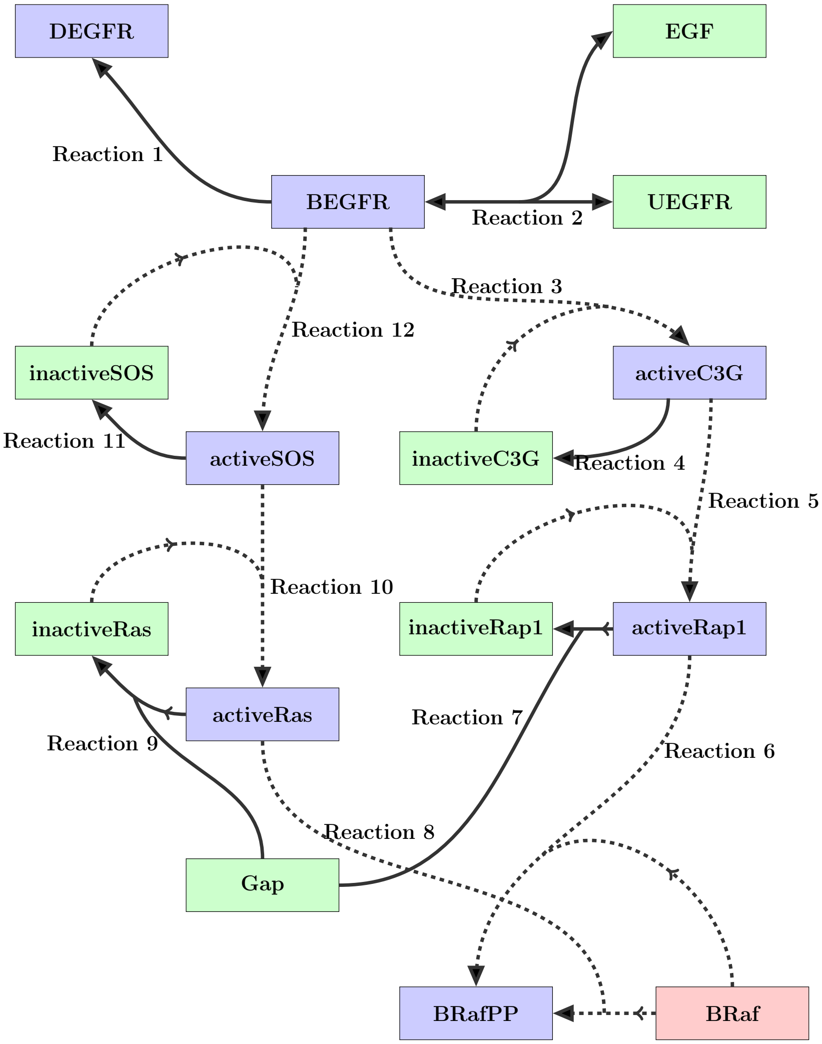
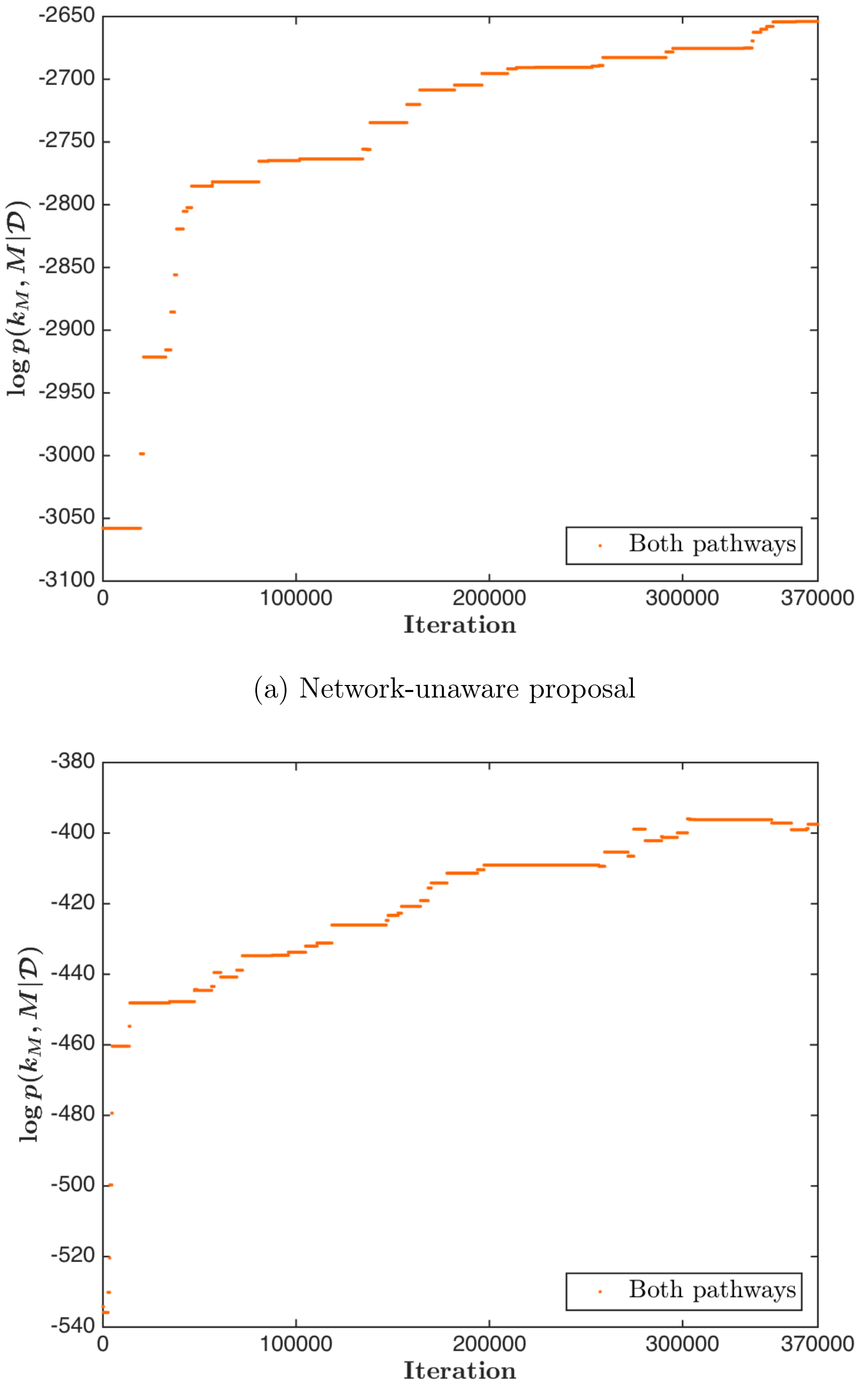


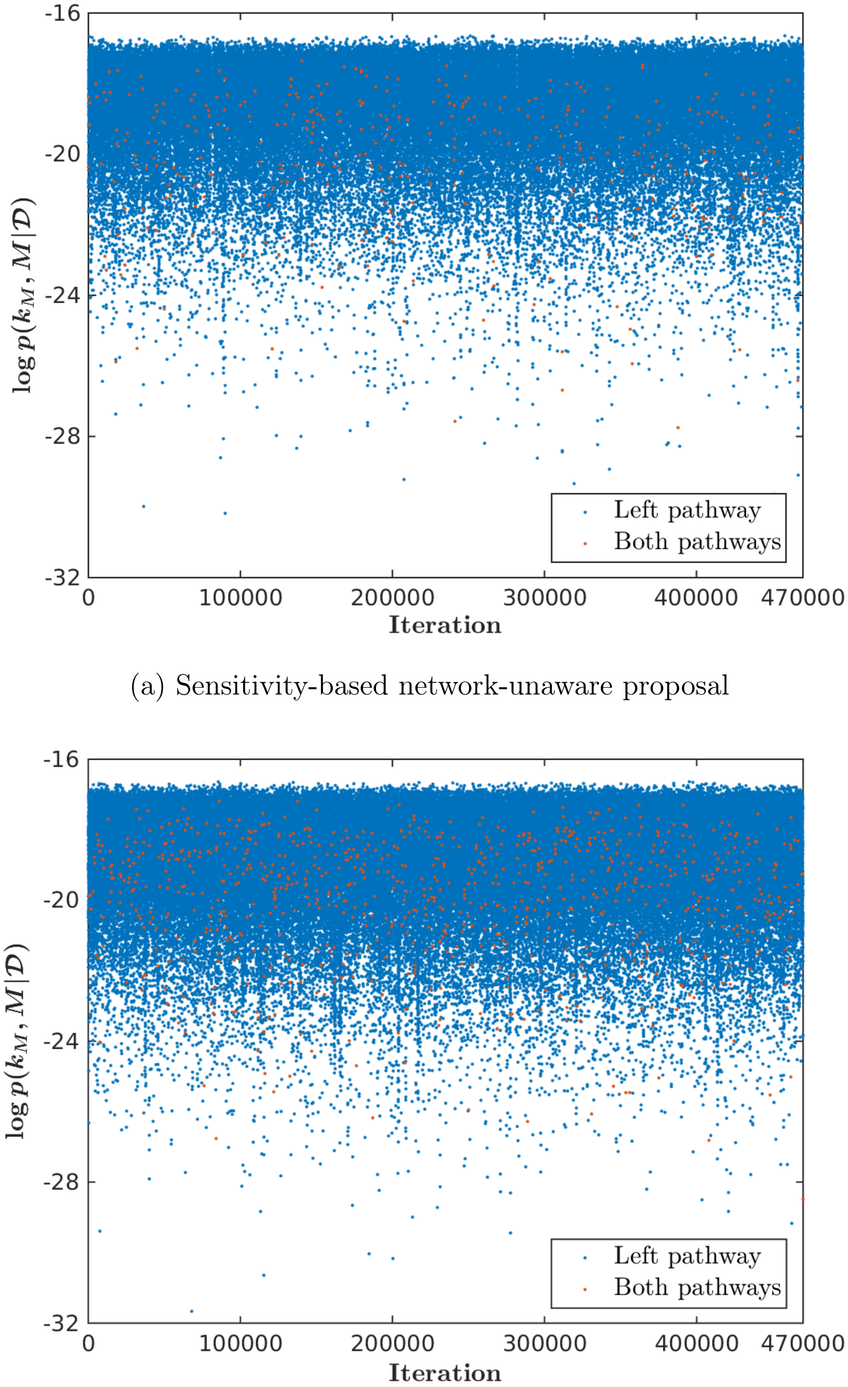
Figure 4-15: A reaction network with 6 (reactions 3, 5, 6, 8, 10, and 12) uncertain reactions. BRaf is the observable.



(a) Network-unaware proposal

(b) Network-aware proposal

Figure 4-16: MCMC trace plots for Example 3 without sensitivity-based proposals: the generated posterior samples are only from models with both pathways



(a) Sensitivity-based network-unaware proposal

(b) Sensitivity-based network-aware proposal

Figure 4-17: MCMC trace plots for Example 3 with sensitivity-based proposals: posterior samples from models with both pathways in orange and samples from models with only the left pathway in blue

	Reaction	$\log_{10} k^{*a}$	k_M^b	Prior uncertainty
1	BEGFR → DEGFR	0.0	-	–
2a	EGF + UEGFR → BEGFR	1.5	-	–
2b	BEGFR → EGF + UEGFR	0.0	-	–
3	inactiveC3G+BEGFR → activeC3G+BEGFR	0.5	3386.3875	$\log_{10} k = \mathcal{N}(1.2, 0.1)$
4	activeC3G → inactiveC3G	2.0	-	–
5	inactiveRap1+activeC3G → activeRap1+activeC3G	2.0	3566	$\log_{10} k = \mathcal{N}(2.7, 0.1)$
6	BRaf+activeRap1 → BRafPP+activeRap1	0.4	17991.179	$\log_{10} k = \mathcal{N}(1.1, 0.1)$
7	activeRap1+Gap → inactiveRap1+Gap	1.0	6808.32	–
8	BRaf+activeRas → BRafPP+activeRas	0.5	7631.63	$\log_{10} k = \mathcal{N}(0.5, 0.1)$
9	activeRas+Gap → inactiveRas+Gap	0.0	12457.816	–
10	inactiveRas+activeSOS → activeRas+activeSOS	0.5	13.73	$\log_{10} k = \mathcal{N}(0.5, 0.1)$
11	activeSOS → inactiveSOS	4.0	9834.13	–
12	inactiveSOS+BEGFR → activeSOS+BEGFR	2.5	8176.56	$\log_{10} k = \mathcal{N}(2.5, 0.1)$

[a] logarithm (base rate constant value)

[b] Base value of Michaelis constant (Obtained from Xu et al. [118])

Table 4.5: Proposed reactions for Example 3

Method†	$p(M)^a$	$\bar{\alpha}_M^b$	$\bar{\alpha}_C^c$	ESS^d	ESS/min^e
Sensitivity-based network unaware	0.022	0.236	0.0035	342	1.921
Sensitivity-based network aware	0.022	0.358	0.0076	932	5.608

†: Performance is averaged over 5 simulation runs

a: Posterior probability of the data-generating model

b: Between-model move acceptance rate

c: Between-cluster move acceptance rate

d: Effective sample size for 10000 samples

e: The absolute values depend on the tolerances chosen for the ODE solver

Table 4.6: Summary statistics of MCMC simulations (Example 3).

terior distribution using our network-aware algorithm (Section 4.3.2) and the network-unaware 2^{nd} order proposal of Brooks et al. [20], which are color coded according to the pathway they belong. Blue points are posterior samples from all models that belong to the left pathway, i.e., models that do not contain any of reactions 3, 5, and 6. Orange points are posterior samples from models that operate with both—left and right branch—pathways, i.e., models that necessarily include reactions 3, 5, 6, 8, 10 and 12. We see that even after 400000 samples are generated, the samplers remain confined to the models with both pathways without ever switching to models belonging to the left pathway. Next, we compare the sampling efficiency of our sensitivity-based network-aware algorithm (Section 4.3.3) to the sensitivity-based network-unaware approach. We simulated 5 replications of 500000 samples using both the approaches. 30000 samples each were discarded as burn-in. All simulations produce similar posterior inferences, thereby indicating convergence. Figure 4-17 shows samples generated from the posterior using the two approaches color coded according to the pathway they belong. The higher frequency of moves between the left pathway models and both pathways models with the network-aware approach is a sign of faster posterior exploration and consequently better MCMC mixing. Table 4.6 shows the acceptance rates of between models moves and between-cluster moves for the two approaches. High model-switching and cluster-switching acceptance rates with the same posterior inference is an indication of superior mixing of the network-aware approach. In Table 4.6, we also present the ESS for the number-of-reactions-in-model statistic. The network-aware scheme has an ESS that is roughly three times ESS of the network-unaware approach. The ESS per minute diagnostic in the last column of Table 4.6 confirms favourability of the network-aware approach with the computational cost taken into account. The absolute value of ESS per minute depends on the relative and absolute tolerance settings of the ODE solver. In particular, we chose very tight tolerances, but higher ESS/min can be obtained with loose tolerances. Nonetheless, the relative values of ESS/min demonstrate the advantage of using the network-aware sampling approach.

Finally, in Figure 4-18, we present the variance of the estimated posterior model probabilities. The variance estimates are calculated based on 5 independent chains. The results of the sensitivity-based network-aware sampler are compared to the sensitivity-based network-unaware approach, with and without derandomization. We can see roughly an order of magnitude lower variance values using our network-aware sampling approach compared to the network-unaware approach with derandomization and two orders of magnitude lower variances compared to the network-unaware approach without derandomization.

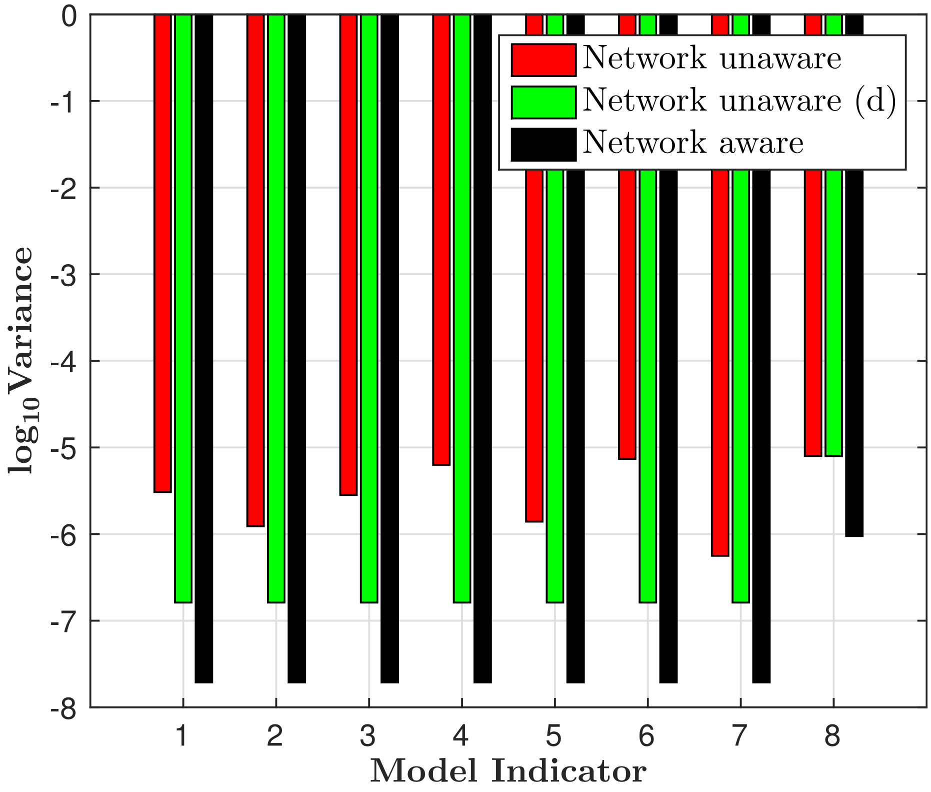


Figure 4-18: Variance of the eight highest posterior model probability estimates in Example 3

4.4.5 Example 4: ten-dimensional nonlinear network inference

Our final example is a large scale nonlinear network inference problem with 10 uncertain reactions. Once again, we consider a protein-signalling network consisting of 15 species and 12 potential species interactions (Figure 4-19). The ODE forward model governing the evolution of species concentrations is described in detail in Appendix B.1. We keep only reactions 1 and 2 fixed (denoted by thick lines in the reaction graph 4-19 and shaded pink in Table 4.7) and thus they are included in all the inferred models. The rate constants of all fixed reactions and Michaelis constants of all reactions are set to their base values (Table 4.7). Reactions 3–12 are uncertain and the concentration of BRaf is again the observable. With the above ten uncertain reactions, the number of potential models is 1024. And with only BRaf as the

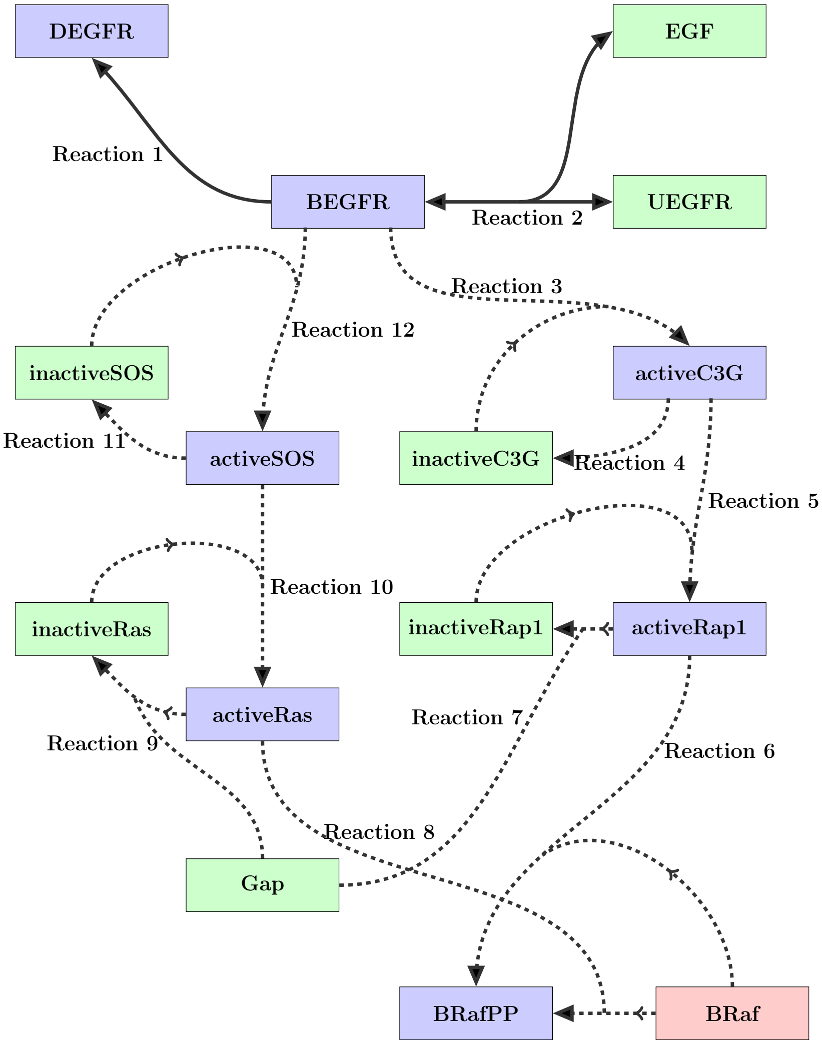


Figure 4-19: A reaction network with 10 (reactions 2–12) uncertain reactions. BRaf is the observable.

observable, the number of clusters is 24.

We generated 30 i.i.d. data points with noise model $\mathcal{N}(0, 0.04)$ and all rate constants and Michaelis constants set to their base values (Table 4.7). We impose independent Gaussian priors on the rate constants of the uncertain reactions with means and variances as shown in Table 4.7. The prior probability distribution over all plausible models is taken to be uniform. The noise variance for the likelihood function, as in the data generating process, is taken as $\sigma^2 = 0.04$.

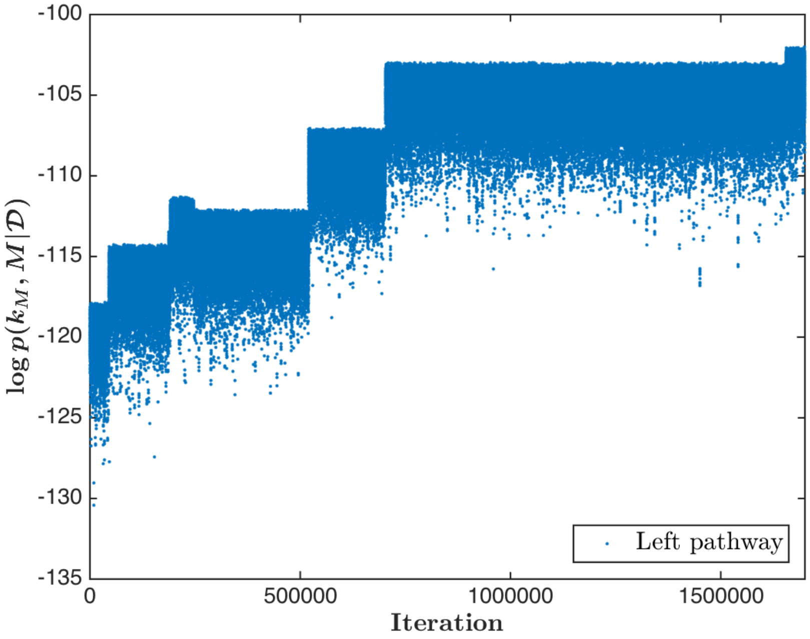
	Reaction	$\log_{10} k^{*a}$	k_M^b	Prior uncertainty
1	BEGFR → DEGFR	0.0	-	–
2a	EGF + UEGFR → BEGFR	1.5	-	–
2b	BEGFR → EGF + UEGFR	0.0	-	–
3	inactiveC3G+BEGFR → activeC3G+BEGFR	0.5	3386.3875	$\log_{10} k = \mathcal{N}(1.2, 0.1)$
4	activeC3G → inactiveC3G	2.0	-	$\log_{10} k = \mathcal{N}(2.0, 0.1)$
5	inactiveRap1+activeC3G → activeRap1+activeC3G	2.0	3566	$\log_{10} k = \mathcal{N}(2.7, 0.1)$
6	BRaf+activeRap1 → BRafPP+activeRap1	0.4	17991.179	$\log_{10} k = \mathcal{N}(1.1, 0.1)$
7	activeRap1+Gap → inactiveRap1+Gap	1.0	6808.32	$\log_{10} k = \mathcal{N}(1.0, 0.01)$
8	BRaf+activeRas → BRafPP+activeRas	0.5	7631.63	$\log_{10} k = \mathcal{N}(0.5, 0.1)$
9	activeRas+Gap → inactiveRas+Gap	0.0	12457.816	$\log_{10} k = \mathcal{N}(0.0, 0.01)$
10	inactiveRas+activeSOS → activeRas+activeSOS	0.5	13.73	$\log_{10} k = \mathcal{N}(0.5, 0.1)$
11	activeSOS → inactiveSOS	4.0	9834.13	$\log_{10} k = \mathcal{N}(4.0, 0.01)$
12	inactiveSOS+BEGFR → activeSOS+BEGFR	2.5	8176.56	$\log_{10} k = \mathcal{N}(2.5, 0.1)$

[a] logarithm (base rate constant value)

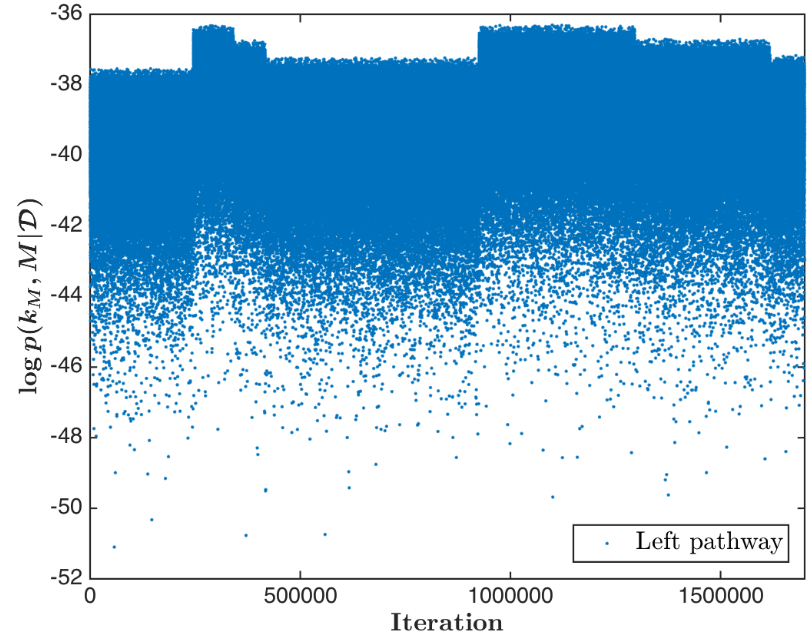
[b] Base value of Michaelis constant (Obtained from Xu et al. [118])

Table 4.7: Proposed reactions for Example 4

Samplers that did not use sensitivity-based proposals show very poor mixing. Figure 4-20 shows samples generated (after a burn-in of 300000 samples) from the posterior distribution using our network-aware algorithm (Section 4.3.2) and the network-unaware 2^{nd} order proposal of Brooks et al. [20], which are color coded according to the pathway they belong. Blue points are posterior samples from all models that belong to the left pathway, i.e., models that do not contain any of reactions 3, 5, and 6. Orange points are posterior samples from models that operate with both—left and right branch—pathways, i.e., models that necessarily include reactions 3, 5, 6, 8, 10 and 12. We see that even after 2 million samples are generated, the samplers remain confined to the models from the left pathway without ever switching to models with both pathways. Next, we compare the sampling efficiency of

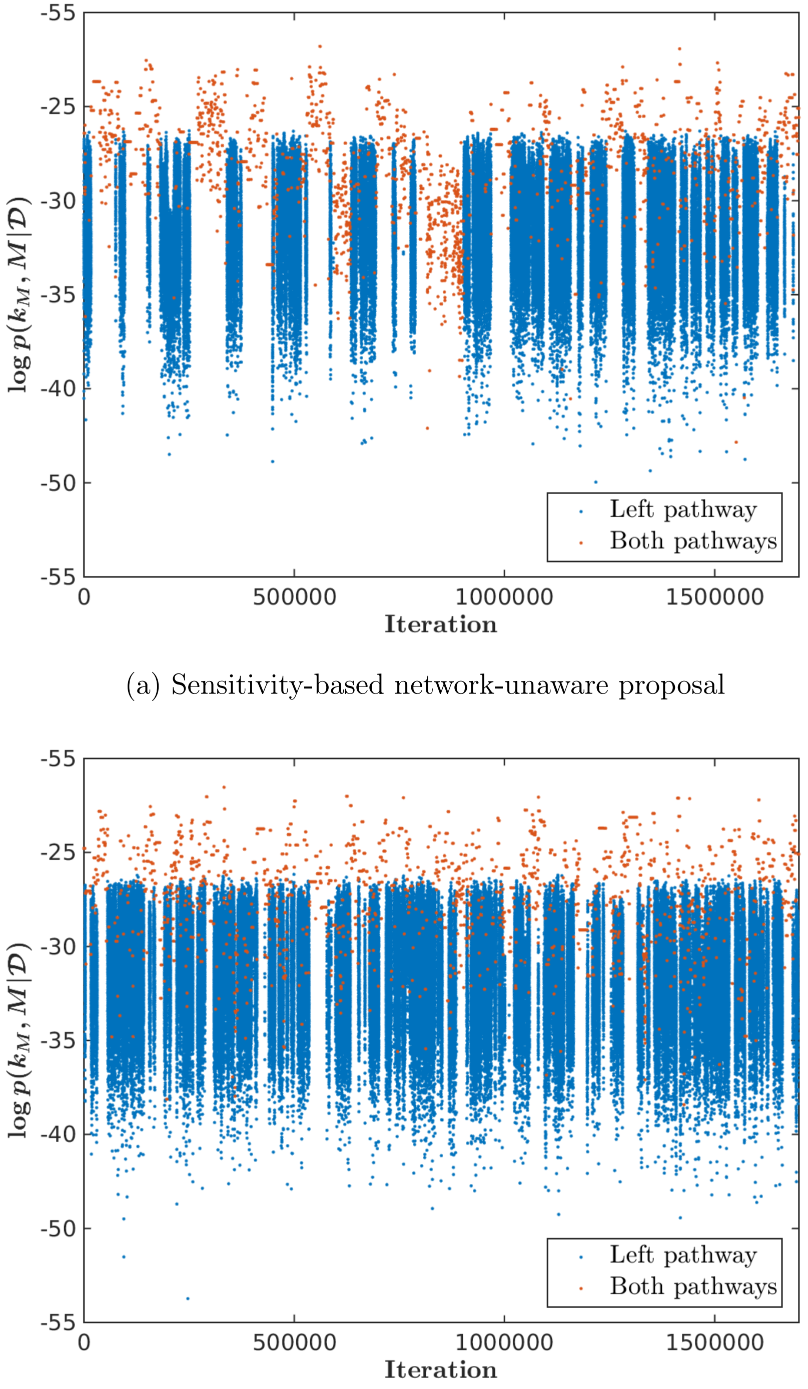


(a) Network-unaware proposal



(b) Network-aware proposal

Figure 4-20: MCMC trace plots for Example 4 without sensitivity-based proposals: the generated posterior samples are only from models with only the left pathway



(a) Sensitivity-based network-unaware proposal

(b) Sensitivity-based network-aware proposal

Figure 4-21: MCMC trace plots for Example 4 with sensitivity-based proposals: posterior samples from models with both pathways in orange and samples from models with only the left pathway in blue

our sensitivity-based network-aware algorithm (Section 4.3.3) to the sensitivity-based network-unaware approach. We simulated 5 replications of 2 million samples using both the approaches. 300,000 samples each were discarded as burn-in. Figure 4-21 shows samples generated from the posterior distribution using the two approaches color coded according to the pathway they belong. We observe that the frequency of moves between the left-pathway models and both-pathways models is higher with the network-aware approach, indicating faster posterior exploration. Table 4.8 shows the acceptance rates of between-model moves and between-pathway moves for the two approaches. High model-switching and pathway-switching acceptance rates with the same posterior inference is an indication of superior mixing of the network-aware approach. In Table 4.8, we also present the ESS for the number-of-reactions-in-model statistic. The network-aware approach has an ESS that is roughly three times the ESS obtained using the network-unaware approach. The ESS per minute diagnostic in the last column of Table 4.8 also supports the use of the network-aware approach.

Method†	$p(M)^a$	$\bar{\alpha}_M^b$	$\bar{\alpha}_P^c$	ESS^d	ESS/min^e
Sensitivity-based network unaware	0.145	0.095	0.0013	86	3.37×10^{-3}
Sensitivity-based network aware	0.157	0.145	0.0027	275	8.66×10^{-3}

†: Performance is averaged over 5 simulation runs
a: Posterior probability of the data-generating model
b: Between-model move acceptance rate
c: Between-pathway move acceptance rate
d: Effective sample size for 1000000 samples
e: The absolute values depend on the tolerances chosen for the ODE solver

Table 4.8: Summary statistics of MCMC simulations (Example 4).

Chapter 5

Network inference with approximation

In Chapters 3 and 4, we presented *exact methods* which converge to the true posterior distribution asymptotically. Many times, however, when the number of plausible models is very large ($> 10^4$), the need for also proposing parameter values for between-model moves could render exact model-space sampling methods very expensive. In this chapter, we propose an approximation-based approach to nonlinear network inference problems. Asymptotic inference methods such as Laplace's method and the Bayesian information criterion are popular alternatives to an exact Bayesian approach [75]. By making assumptions about the structure of parameter posterior distribution, these methods allow computationally efficient Bayesian model inference. Specifically, the model evidence is approximated by solving a high-dimensional optimization problem and evaluating a Hessian matrix for each model. If, however, the number of models is very high, the enumeration of model evidences for all models is again prohibitively expensive. We propose simulating a Markov chain Monte Carlo algorithm with only the model indicator M_i in the state space and for each visited model approximating the model evidence using Laplace's method. For linear models with conjugate priors, the Laplace approximation produces exact evidence, and

MCMC over model indicators is often used for variable selection and inference of graphical models [83]. For general non-conjugate models, approximations in MCMC simulations have been used in the past [32, 34, 61]. In particular, Guihenneuc et al. [61] provide theory on the error induced in posterior distributions while employing Laplace approximations in MCMC simulations.

The inference of reaction networks governed by nonlinear species interactions (thus nonlinear parameter-to-observable maps) with limited and sparse experimental data leads to posterior distributions that are multimodal. The standard single-chain MCMC approach when used on multimodal posteriors has a tendency to get stuck in one of the modes. We extend the above approximation-based network inference approach to a population-based MCMC scheme, which involves running parallel Markov chains that exchange information among themselves to ensure all posterior modes are adequately traversed. We show the superiority of the population-based scheme in exploring a posterior distribution over a space of $O(10^9)$ reaction networks in comparison to a single chain simulation. This method allows systematic exploration of posterior distributions over nonlinear networks when exact sampling of networks is infeasible.

5.1 Laplace's method

Recall the application of Bayes' rule to models $\{M_i\}$ gives

$$p(M_i|\boldsymbol{\mathcal{D}}) = \frac{p(\boldsymbol{\mathcal{D}}|M_i)p(M_i)}{p(\boldsymbol{\mathcal{D}})}. \tag{5.1}$$

Here, $p(M_i|\boldsymbol{\mathcal{D}})$ is the posterior probability of M_i conditioned on data $\boldsymbol{\mathcal{D}}$; $p(\boldsymbol{\mathcal{D}}|M_i)$, also called the model evidence, is the marginal likelihood of observing $\boldsymbol{\mathcal{D}}$; and $p(M_i)$ is the prior probability of model M_i. Specifically,

$$p(\boldsymbol{\mathcal{D}}|M_i) = \int p(\boldsymbol{\mathcal{D}}|\boldsymbol{k_i}, M_i)p(\boldsymbol{k_i}|M_i)d\boldsymbol{k_i}, \tag{5.2}$$

where $\boldsymbol{k_i}$ is a model-specific multidimensional parameter. Laplace's method approximates (5.2) by computing a Gaussian approximation to the posterior density. Let the joint density of $\boldsymbol{\mathcal{D}}$ and $\boldsymbol{k_i}$ conditioned on model M_i be $l(\boldsymbol{k_i})$: $l(\boldsymbol{k_i}) = \log(p(\boldsymbol{\mathcal{D}}|\boldsymbol{k_i}, M_i)p(\boldsymbol{k_i}|M_i))$. The Laplace's method involves finding the posterior mode:

$$\tilde{\boldsymbol{k}}_{\boldsymbol{i}} = \arg\max_{\boldsymbol{k_i}} \log(p(\boldsymbol{\mathcal{D}}|\boldsymbol{k_i}, M_i)p(\boldsymbol{k_i}|M_i)), \tag{5.3}$$

and computing a second-order Taylor expansion of $l(\boldsymbol{k_i})$ about $\tilde{\boldsymbol{k}}_{\boldsymbol{i}}$. Exponentiating the quadratic yields an approximation to $p(\boldsymbol{\mathcal{D}}|\boldsymbol{k_i}, M_i)p(\boldsymbol{k_i}|M_i)$, which has the form for a normal density with mean $\tilde{\boldsymbol{k}}_{\boldsymbol{i}}$ and covariance matrix $\tilde{\boldsymbol{\Sigma}} = (-\nabla^2 l(\tilde{\boldsymbol{k}}_{\boldsymbol{i}}))^{-1}$. Integrating the approximation gives

$$\tilde{p}(\boldsymbol{\mathcal{D}}|M_i) = (2\pi)^{d/2}|\tilde{\boldsymbol{\Sigma}}|^{1/2}p(\boldsymbol{\mathcal{D}}|\boldsymbol{k_i}, M_i)p(\tilde{\boldsymbol{k}}_{\boldsymbol{i}}|M_i), \tag{5.4}$$

where d is the dimensionality of $\boldsymbol{k_i}$. Kass et al. [76] give conditions under which as $n_{data} \to \infty$, $p(\boldsymbol{\mathcal{D}}|M_i) = \tilde{p}(\boldsymbol{\mathcal{D}}|M_i)(1 + O(n_{data}^{-1}))$. Plugging 5.4 into 5.5 gives the approximate posterior model probability:

$$\tilde{p}(M|\boldsymbol{\mathcal{D}}) = \frac{\tilde{p}(\boldsymbol{\mathcal{D}}|M)p(M)}{p(\boldsymbol{\mathcal{D}})}. \tag{5.5}$$

5.2 Large-scale approximate model inference

The systematic application of Laplace's method to all models when the number of models is very high is computationally prohibitive. Using a Markov chain Monte Carlo scheme to explore the approximate posterior distribution over models provides a simulation-consistent estimate of approximate model posteriors. Thus, we simulate $M^n \sim \tilde{p}(M|\boldsymbol{\mathcal{D}})$ to get estimates of posterior model probabilities as

$$\hat{p}(M_i|\boldsymbol{\mathcal{D}}) = \frac{1}{N}\sum_{n=1}^{N} \mathbb{I}_{M_i}(M^n). \tag{5.6}$$

Since each model has a network structure and the data may be sparse, we have

$$\begin{aligned} p(\boldsymbol{\mathcal{D}}|M_m) &= \int\cdots\int p(\boldsymbol{\mathcal{D}}|\boldsymbol{k_i}, M_i)p(\boldsymbol{k_i}|M_i)d\boldsymbol{k_i} \\ &= \int\cdots\int p(\boldsymbol{\mathcal{D}}|\boldsymbol{k_{e,i}}, M_{e,i})p(\boldsymbol{k_{e,i}}|M_{e,i})d\boldsymbol{k_{e,i}}, \end{aligned}$$

where $M_{e,i}$ is the effective network corresponding to M_i and $\boldsymbol{k_{e,i}}$ are the parameters of the effective network (Chapter 4). Thus, we apply Laplace's method directly to the effective networks. The sequence of steps in large-scale approximate model inference are given in Algorithm 9.

Algorithm 9 Approximate model inference MCMC

```
Given: A set of models $\{M_i\}$ with corresponding parameter vectors $\{\boldsymbol{k_i}\}$, likelihood functions $\{p(\boldsymbol{\mathcal{D}}|M_i, \boldsymbol{k_i})\}$, parameter prior densities $\{p(\boldsymbol{k_i})\}$, and model prior distribution $p(M)$.
Initlialize starting point $(M^0, \boldsymbol{k}_{M^0})$; $M_e^0 = \mathit{eff}(M^0)$
for $n = 0$ to $N_{iter}$ do
    Sample $M' \sim q(M'|M^n = M)$; $M_e' = \mathit{eff}(M')$
    Determine approximate model evidence $\tilde{p}(\boldsymbol{\mathcal{D}}|M_e')$ of $M_e'$
    Sample $p \sim \mathcal{U}_{[0,1]}$
    if $p < A(M, M') = \min\{1, \frac{\tilde{p}(\boldsymbol{\mathcal{D}}|M_e')p(M')q(M^n|M')}{\tilde{p}(\boldsymbol{\mathcal{D}}|M_e^n)p(M^n)q(M'|M^n)}\}$ then
        $M^{n+1} = M'$
    else
        $M^{n+1} = M^n$
    end if
end for
```

5.2.1 Setup of the Bayesian model inference problem

Likelihood function

We employ an i.i.d. additive Gaussian model for the difference between model predictions and observations; thus the data are represented as

$$\boldsymbol{\mathcal{D}} = \boldsymbol{G}(\boldsymbol{k}) + \boldsymbol{\epsilon_n}, \tag{5.7}$$

where $\boldsymbol{\epsilon_n} \sim \mathcal{N}_n(\boldsymbol{0}, \sigma^2 \boldsymbol{I_n})$, n is the number of observations, $\boldsymbol{I_n}$ is an n-by-n identity matrix, and $\boldsymbol{G}(\boldsymbol{k})$ is the prediction of the forward model at the given value of the reaction parameters. The specific values of the noise standard deviations σ are given later. The deterministic predictions $\boldsymbol{G}(\boldsymbol{k})$ are obtained with the ODE integrator. The likelihood function is thus given by

$$\begin{aligned} p(\boldsymbol{\mathcal{D}}|\boldsymbol{k}) &= \mathcal{N}_n(\boldsymbol{\mathcal{D}}|\boldsymbol{G}(\boldsymbol{k}), \sigma^2 \boldsymbol{I_n}) \\ &= \prod_{t=1}^{n} \mathcal{N}(d^t | G_t(\boldsymbol{k}), \sigma^2) \\ &= \prod_{t=1}^{n} \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(d^t - G_t(k))^2}{2\sigma^2} \right), \end{aligned} \tag{5.8}$$

where d^t are components of the data vector $\boldsymbol{\mathcal{D}}$.

Prior specification

Since reaction rate constants must be positive, while their uncertainties may have multiple orders of magnitude, we take the prior distribution to be a lognormal distribution on the rate constant. Specifically, we set each $p(k_i)$ to

$$p(k_i) : \log_{10} k_i \sim \mathcal{N}(\log_{10} k_i^*, \sigma_i^2), \tag{5.9}$$

where each k_i^* above is the base value of the ith rate constant.

5.2.2 Example 1: 10 dimensional reaction network

We demonstrate the large-scale approximate model inference approach on a 10 dimensional example with synthetic data. Consider the reaction network shown in Figure 5-1. The set of reactions shown in this network are a subset of a larger model proposed for the activation of the extracellular signal-regulated kinase (ERK) pathway by epidermal growth factor [118]. The ODE forward model governing the evolution of species concentrations is described in detail in Appendix B.1. We keep reactions 1 and 2 fixed (denoted by thick lines in the reaction grap 5-1 and shaded pink in Table 5.1) and thus they are included in all the inferred models. The rate constants of all fixed reactions and Michaelis constants of all reactions are set to their base values (Table 5.1). Reactions 3–12 are taken to be uncertain and the concentration of BRaf is taken to be the observable. With the above ten uncertain reactions, the number of potential models is 1024. And with only BRaf as the observable, the number of clusters is 24.

	Reaction	$\log_{10} k^{*a}$	k_M^b	Prior uncertainty
1	BEGFR $\rightarrow$ DEGFR	0.0	-	–
2a	EGF + UEGFR $\rightarrow$ BEGFR	1.5	-	–
2b	BEGFR $\rightarrow$ EGF + UEGFR	0.0	-	–
3	inactiveC3G+BEGFR $\rightarrow$ activeC3G+BEGFR	0.5	3386.3875	$\log_{10} k = \mathcal{N}(1.2, 0.1)$
4	activeC3G $\rightarrow$ inactiveC3G	2.0	-	$\log_{10} k = \mathcal{N}(2.0, 0.1)$
5	inactiveRap1+activeC3G $\rightarrow$ activeRap1+activeC3G	2.0	3566	$\log_{10} k = \mathcal{N}(2.7, 0.1)$
6	BRaf+activeRap1 $\rightarrow$ BRafPP+activeRap1	0.4	17991.179	$\log_{10} k = \mathcal{N}(1.1, 0.1)$
7	activeRap1+Gap $\rightarrow$ inactiveRap1+Gap	1.0	6808.32	$\log_{10} k = \mathcal{N}(1.0, 0.1)$
8	BRaf+activeRas $\rightarrow$ BRafPP+activeRas	0.5	7631.63	$\log_{10} k = \mathcal{N}(0.5, 0.1)$
9	activeRas+Gap $\rightarrow$ inactiveRas+Gap	0.0	12457.816	$\log_{10} k = \mathcal{N}(0.0, 0.01)$
10	inactiveRas+activeSOS $\rightarrow$ activeRas+activeSOS	0.5	13.73	$\log_{10} k = \mathcal{N}(0.5, 0.1)$
11	activeSOS $\rightarrow$ inactiveSOS	4.0	9834.13	$\log_{10} k = \mathcal{N}(4.0, 0.01)$
12	inactiveSOS+BEGFR $\rightarrow$ activeSOS+BEGFR	2.5	8176.56	$\log_{10} k = \mathcal{N}(2.5, 0.1)$

[a] logarithm (base rate constant value)

[b] Base value of Michaelis constant (Obtained from Xu et al. [118])

Table 5.1: Proposed reactions for Example 1

We simulate 30 i.i.d. data points with additive noise model $\mathcal{N}(0, 4)$ and all rate constants and Michaelis constants set to their base values (Table 5.1). We impose independent Gaussian priors on the logarithm of rate constants of the uncertain reactions with means and variances as shown in Table 5.1. The prior probability dis-

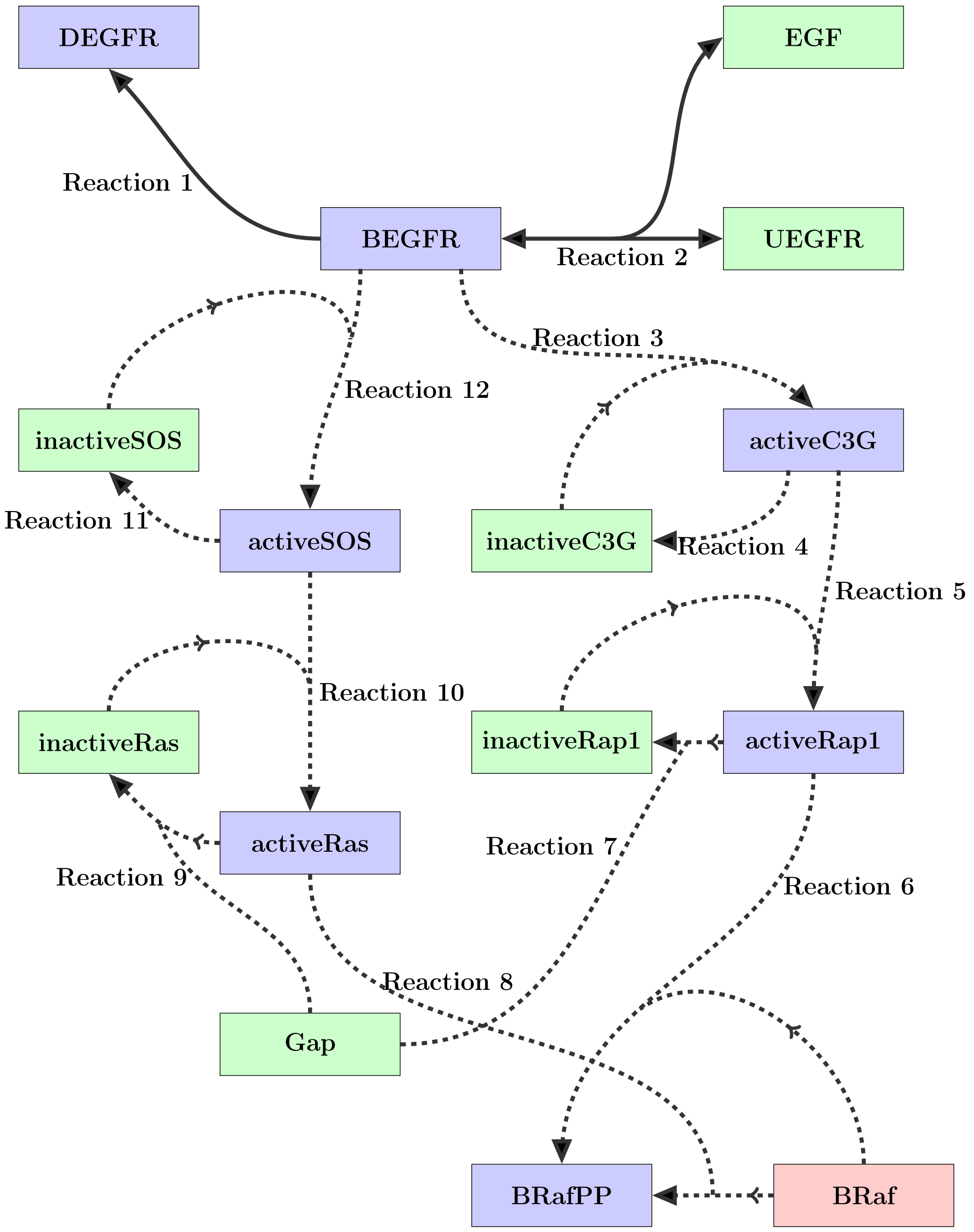


Figure 5-1: Reaction network of Example 1

tribution over all plausible models is taken to be uniform. The noise variance for the likelihood function, as in the data generating process, is taken as $\sigma^2 = 4$.

	Posterior probabilities	
Effective network	Enumeration	Sampling†
1,2,3,5,6,7,8,10,12	0.435	0.434
1,2,3,4,5,6,7,8,10,12	0.463	0.465
1,2,3,5,6,7,8,10,11,12	0.101	0.101

$\dagger$: Averaged over 3 MCMC replications

Table 5.2: Summary of posterior probabilites for Example 1 by enumeration and sampling

We generate 1 million samples using the approximate posterior Markov chain Monte Carlo algorithm and in Table 5.2 show that the posterior probabilities agree with the values obtained by enumeration of model evidences of all plausible models. The enumeration of model evidences involves computing an approximate model evidence using the Laplace's method for each model. The example presented here has a space of 2^{10} models. In the following sections, we present examples with 2^{30} models, for which the exhaustive evidence calculation is infeasible. As we scale the dimensionality of the problem, we further find that the posterior distribution for the nonlinear network inference problem are multimodal. To explore the multimodal posterior, we have developed the population-based approximate model inference algorithm.

5.2.3 Consistency of approximate model inference

Bayesian model selection (BMS) is consistent, i.e., given enough data and the data-generating model among the set of models being compared, the true model is selected by BMS [12]. The methods presented in this chapter construct an approximation of the evidence and thus estimate approximate posterior model probabilities. However, since the error of the Laplace's method goes down as $O(n_{data}^{-1})$, the consistency of BMS can be expected to be retained by the approximate posterior based model selection method. To investigate the consistency of the approximate model inference algorithm, we evaluate posterior model probabilities for the networks considered in Example 1 with steadily increasing data sets. Note Example 1 infact compares the same set of networks as in Example 4 of Chapter 4, but with a different noise model. In order to also compare the result we obtained using the approximate model inference approach to the exact sampling approach of Chapter 4, we perform the consistency study with the noise model ($\sigma^2 = 0.04$) employed in Chapter 4. The data in each case are generated with a network consisting of all proposed reactions. The details of the size of the data sets and the obtained posterior probabilities are presented in Table 5.3. Firstly, we find that the results obtained for the data set with 30 points are very different using the approximate model inference and exact sampling approaches. Specifically, we find that the posterior probability of the data-generating model (effective network with reactions 1–12) using the approximate model inference approach is ≈ 0.0, whereas in Chapter 4 it was ≈ 0.15. In addition all the posterior mass is concentrated over models with both the left and right pathways included in the approximate approach. In contrast in Chapter 4, we found that models that only contain the left pathway of the reaction network have sizable probability. This discrepancy between exact sampling and approximate inference results is not surprising, since the inference results with the two methods can be quite different with a small data set. As the amount of data increases, however, we find that the posterior probability of the data-generating model increases and ultimately converges to 1. Thus the data-generating model is selected by the approximate model inference approach given sufficient data.

5.3 Population-based Markov chain Monte Carlo

Posterior distributions with multiple modes arise in many areas of applied statistics, e.g., gene clustering [66] and population genetics [101], and their exploration is well known to be a challenging sampling problem. To define multimodality in the present context of network inference, one needs to first introduce a notion of distance metric.

	Posterior probabilities					
Effective network	$n_{data} = 30$		$n_{data} = 880$		$n_{data} = 3840$	
1,2,3,5,6,8+	Sampling	Enumeration	Sampling	Enumeration	Sampling	Enumeration
7,9,10,11,12	0.013	0.013	0.000	0.000	0.000	0.000
7,10,11,12	0.087	0.089	0.000	0.000	0.000	0.000
9,10,11,12	0.163	0.162	0.000	0.000	0.000	0.000
9,10,12	0.098	0.098	0.000	0.000	0.000	0.000
10,11,12	0.238	0.236	0.000	0.000	0.000	0.000
10,12	0.083	0.082	0.000	0.000	0.000	0.000
7,9,10,12	0.011	0.012	0.001	0.001	0.000	0.000
7,10,12	0.306	0.308	0.022	0.025	0.000	0.000
4,7,10,12	0.000	0.000	0.002	0.002	0.000	0.000
4,7,10,11,12	0.000	0.000	0.004	0.005	0.000	0.000
4,7,9,10,12	0.000	0.000	0.156	0.159	0.001	0.001
4,7,9,10,11,12	**0.000**	**0.000**	**0.808**	**0.814**	**0.999**	**0.999**

Table 5.3: Consistency of posterior probability estimates

A natural measure of distance between networks is the number of distinct reactions in the networks. Thus two networks with a small number of distinct reactions may be considered “close” to each other, whereas two networks with a large number of distinct reactions would be “far” from each other. What we find in nonlinear network inference is that in many examples, the posterior distribution is concentrated over models that are separated by a large distance—posterior distribution is multimodal. Using the standard approach of proposing “local” moves in model space—moves between nearby models—in such cases leads to very poor posterior exploration. In particular, the sampler often fails to explore all posterior modes.

Population-based Markov chain Monte Carlo methods provide an approach for efficient exploration of multimodal distributions. These methods involve running parallel MCMC chains over a sequence of closely related distributions and exchanging information between the chains as samples are generated. Geyer [50] first introduced a population-based MCMC method known as parallel tempering in which the sequence of distributions included the target posterior distribution and tempered versions of the posterior distribution. The central idea is that the more tempered distributions are able to traverse the multimodal sample space easily and by exchanging samples between the chains, the chain corresponding to the target distribution is able to jump between disparate modes. A similar approach was also simultaneously proposed in the physics literature by Hukushima et al [70]. Jasra et al. [72] review recent developments in population-based methods. We give here a brief overview of general population-based MCMC methods.

In order to sample from the posterior distribution $p(\boldsymbol{\theta}|\boldsymbol{\mathcal{D}})$, a new target distribution is defined in population-based MCMC as

$$p(\boldsymbol{\theta_{1:S}}) = \prod_{j=1}^{S} p_j(\boldsymbol{\theta_j}), \tag{5.10}$$

where at least one of $p_j(\boldsymbol{\theta_j})$ is taken as $p(\boldsymbol{\theta}|\boldsymbol{\mathcal{D}})$. By designing a transition kernel that is $p(\boldsymbol{\theta_{1:S}})$-irreducible, aperiodic and has $p(\boldsymbol{\theta_{1:S}})$ as the invariant distribution, a population-based MCMC method is constructed. The method consists of a population of chains, where each chain corresponds to one of the distributions $p_j(\boldsymbol{\theta_j})$ and the state of the chains updated according to a set of possible move types. Many different choices of the sequence of distributions $p_j(\boldsymbol{\theta_j})$ can be made. We describe two common choices; see [72] for a complete discussion.

5.3.1 Sequence of distributions

Identical

One choice of the sequence of distributions is to take each of them to be the target posterior distribution $p(\boldsymbol{\theta}|\boldsymbol{\mathcal{D}})$. Although simple, the effectivity of this sequence relies on the original single chain sampler being efficient in exploring the target distribution. This approach was used by Warnes et al. [115], where they update the state of the chains according to a proposal that is a Gaussian approximation of the target distribution constructed with samples from all the chains. Robert and Casella [105] note that this approach may not work well for high-dimensional targets due to poor density estimation in high dimensions.

Tempered

The tempered approach consists of taking the sequence of distributions to be tempered versions of the target distribution [81]. Thus, one possible choice is to take

$$p_j(\boldsymbol{\theta_j}) \propto p(\boldsymbol{\theta}|\boldsymbol{\mathcal{D}})^{\frac{1}{T_j}} p(\boldsymbol{\theta}), \tag{5.11}$$

where $\{T_j\}$ is interpreted as a temperature ladder. The idea is that as the temperature rises, the corresponding distribution becomes less peaky and the respective chain can traverse the space faster. By defining a large number distributions in the sequence, the chains at higher temperatures explore the state space efficiently and by exchanging information between chains, the chains at lower temperatures can jump between distinct modes. Note one of the chains always has $T_j = 1$ and corresponds to the target distribution. The choice of the temperature schedule ($\zeta_j = \frac{1}{T_j}$) is discussed in Jasra et al [72].

5.3.2 Population moves

At each step of the simulation, we choose among a set of possible population moves. Move types ranging from simple Metropolis-Hastings to more non-standard moves such as exchanging samples can performed and we give details of these here.

Mutation

The mutation population move consists of selecting one of the distributions $p_j(\boldsymbol{\theta_j})$ and performing a Metropolis-Hastings move on that distribution. Formally, a chain $t \in \{1, 2, ..., S\}$ is selected, a proposal $\boldsymbol{\theta'_t}$ is made according to a proposal distribution $q_t(\boldsymbol{\theta'_t}|\boldsymbol{\theta_t})$, and the proposal is accepted with probability:

$$\alpha_{prob}(\boldsymbol{\theta'}|\boldsymbol{\theta}) = \min\left\{1, \frac{p_t(\boldsymbol{\theta'_t})q_t(\boldsymbol{\theta_t}|\boldsymbol{\theta'_t})}{p_t(\boldsymbol{\theta_t})q_t(\boldsymbol{\theta'_t}|\boldsymbol{\theta_t})}\right\}, \tag{5.12}$$

Exchange

The Exchange population move is one way to swap information between chains. The move consists of selecting two chains at random and proposing an exchange of the states of the two chains. Say two chains t_1 and t_2 are selected with a uniform probability, their state values $\boldsymbol{\theta_{t_1}}$ and $\boldsymbol{\theta_{t_2}}$ are exchanged with probability:

$$\alpha_{prob}(\boldsymbol{\theta'}|\boldsymbol{\theta}) = \min\left\{1, \frac{p_{t_1}(\boldsymbol{\theta_{t_2}})p_{t_2}(\boldsymbol{\theta_{t_1}})}{p_{t_1}(\boldsymbol{\theta_{t_1}})p_{t_2}(\boldsymbol{\theta_{t_2}})}\right\}. \tag{5.13}$$

5.4 Population-based approximate model inference

We now proceed to explain our population-based approximate model inference MCMC approach for large-scale network inference. With limited and oftentimes sparse available data, the large-scale approximate nonlinear network inference by Algorithm 9 was seen to show poor mixing. The investigation of the reason for poor mixing revealed that the posterior was almost always multimodal. The use of our population-based approximate model inference approach was able to resolve the difficulties and explore the approximate posterior distrbution efficiently.

The population-based approximate model inference scheme consists of expanding the state space to a product of the original target space $\{M_i\}$ and defining a sequence of tempered posterior distributions on $\{M_i\}_{1:S}$. Thus, the population target distribution is taken as

$$p_{1:S}(\boldsymbol{M_{1:S}}) = \prod_{j=1}^{S} p_j(\theta_j), \tag{5.14}$$

where $\theta_j = M$(model indicator) and we take $p_j(\theta_j) \propto \tilde{p}(\boldsymbol{\mathcal{D}}|M)^{\zeta_j}p(M)$. The guidelines for appropriate number of distributions and the temperature schedule ζ_j are discussed in Jasra et al. [72] and we adopt those in our simulations. The population-based approximate model inference here extends the population-based MCMC of Section 5.3 to approximate model posteriors. The steps of the population-based approximate

Algorithm 10 Population-based approximate model inference MCMC

```
Given: A set of models $\{M_i\}$ with corresponding parameter vectors $\{\boldsymbol{k_i}\}$, likelihood functions $\{p(\mathcal{D}|M_i, \boldsymbol{k_i})\}$, parameter prior densities $\{p(\boldsymbol{k_i})\}$, and model prior distribution $p(M)$.
Define a population (of size S) distribution over models: $p_{1:S}(\boldsymbol{M_{1:S}}) = \prod_{j=1}^{S} p_j(M_j)$, where
    $$p_j(M_j) \propto \tilde{p}(\boldsymbol{\mathcal{D}}|M)^{\zeta_j} p(M).$$
eff(M): effective network
Initlialize starting point $(\boldsymbol{M^0_{1:S}}, \boldsymbol{k_{M^0_{1:S}}})$; $\boldsymbol{M^0_{e,1:S}}$=eff$(\boldsymbol{M^0_{1:S}})$
for $n = 0$ to $N_{iter}$ do
    $p \sim \mathbb{U}[0,1]$
    if $p \leq 0.5$ then
        Mutation move:
        Select a chain $t$ to update
        Sample $M'_t \sim q(M'_t|M_t)$; $M'_{e,t}$=eff$(M'_{e,t})$
        Determine approximate model evidence $\tilde{p}(\boldsymbol{\mathcal{D}}|M'_{e,t})$ of $M'_{e,t}$
        Sample $pp \sim \mathbb{U}[0,1]$
        if $pp \leq \min\{1, \frac{\tilde{p}(\boldsymbol{\mathcal{D}}|M'_{e,t})^{\zeta_t} p(M'_t) q(M^n_t|M'_t)}{\tilde{p}(\boldsymbol{\mathcal{D}}|M_{e,t})^{\zeta_t} p(M_t) q(M'_t|M^n_t)}\}$ then
            $M^{n+1}_t = M'_t$
        else
            $M^{n+1}_t = M^n_t$
        end if
    else
        Exchange move:
        Select two chains $t_1$ and $t_2$ uniformly
        Sample $pp \sim \mathbb{U}[0,1]$
        if $pp \leq \min\{1, \frac{\tilde{p}(\boldsymbol{\mathcal{D}}|M^n_{e,t_2})^{\zeta_{t_1}} \tilde{p}(\boldsymbol{\mathcal{D}}|M^n_{e,t_1})^{\zeta_{t_2}}}{\tilde{p}(\boldsymbol{\mathcal{D}}|M^n_{e,t_1})^{\zeta_{t_1}} \tilde{p}(\boldsymbol{\mathcal{D}}|M^n_{e,t_2})^{\zeta_{t_2}}}\}$ then
            $M^{n+1}_{t_1} = M^n_{t_2}$ and $M^{n+1}_{t_2} = M^n_{t_1}$
        else
            $M^{n+1}_{t_1} = M^n_{t_1}$ and $M^{n+1}_{t_2} = M^n_{t_2}$
        end if
    end if
end for
```

model inference method are detailed in Algorithm 10. Again focusing on the network inference problem with possibly sparse data, we calculate approximate evidence by first determining the effective network and then apply the Laplace's method on the parameters of the effective network.

5.4.1 Example 2: 30 dimensional reaction network with a single species observable

We demonstrate our population-based approximate model inference approach on an example with 30 proposed reactions. Consider the reaction network shown in Figure 5-2. The set of reactions shown in this network are the union of all reactions considered to explain the activation of extracellular signal-regulated kinase (ERK) by epidermal growth factor by Xu et al [118]. The concentration of ERK taken as the observable. We generate 30 i.i.d. data points with the noise model $\mathcal{N}(0,4)$ and all rate constants and Michaelis constants set to their base values (Table 5.4). We impose independent Gaussian priors on the logarithm of the rate constants of the uncertain reactions with means and variances as shown in Table 5.4. The prior probability distribution over all plausible models is taken to be uniform. The noise variance for the likelihood function, as in the data generating process, is taken as $\sigma^2 = 4$.

We generate 1 million samples using the single-chain approximate model inference and the population-based approximate inference algorithms. The proposal distribution for the single-chain sampler is taken to be a Poisson distribution Pois(1.0) for the number of reactions r to be added/deleted, coupled with a uniform selection of the r reactions to be added/deleted. In case of the population-based algorithm we take a population of 40 chains and the sequence of distributions corresponding to the 40 chains given by:

$$p_j(\theta_j) \propto \tilde{p}(\boldsymbol{\mathcal{D}}|M)^{\zeta_j} p(M), \tag{5.15}$$

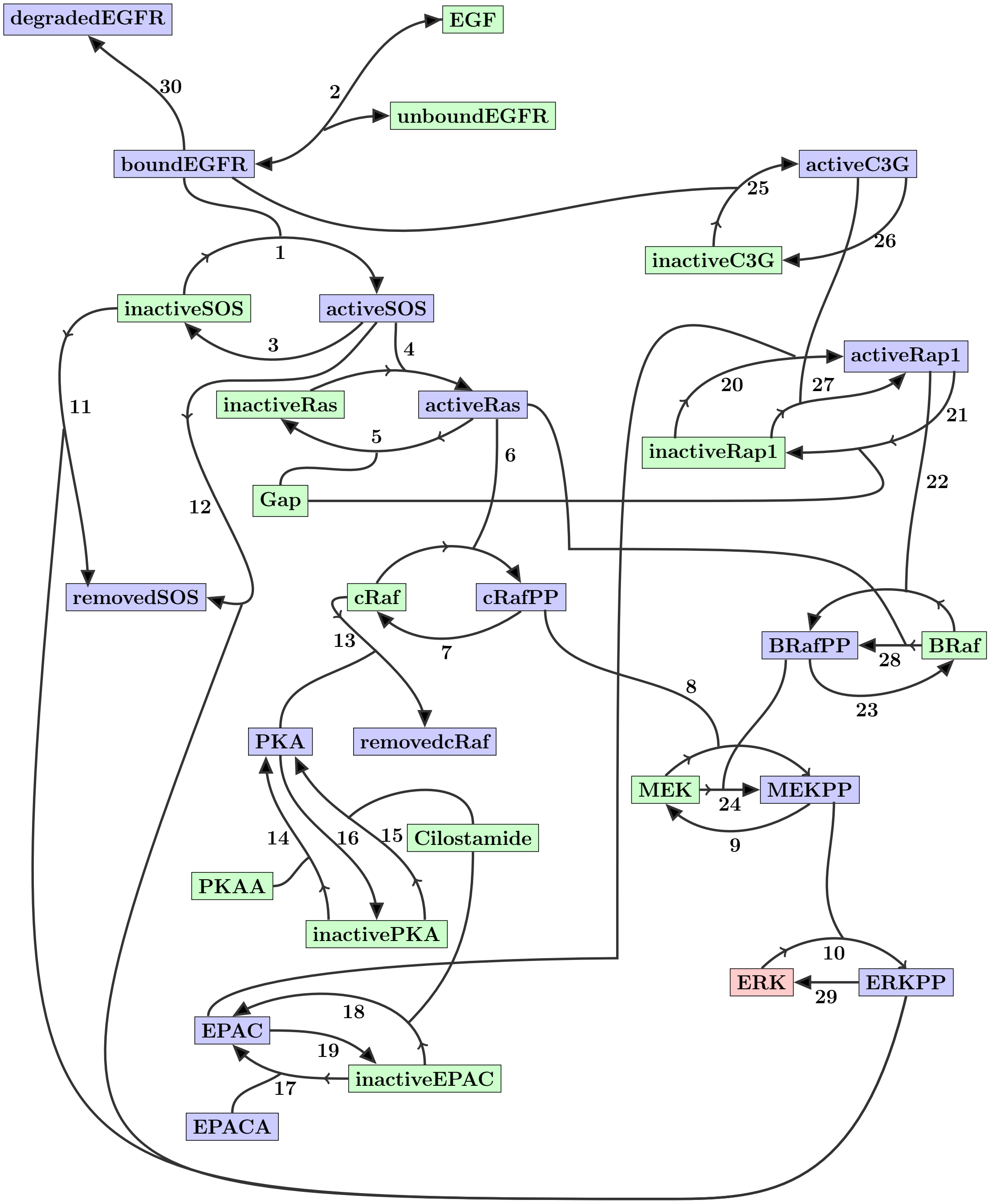


Figure 5-2: Reaction network of Example 2

	Reaction	$\log_{10} k^{*a}$	k_M^b	Prior uncertainty
1	inactiveSOS + boundEGFR → activeSOS + boundEGFR	1.0	1.0	$\log_{10} k = \mathcal{N}(1.0, 0.1)$
2a	EGF+unboundEGFR → boundEGFR	1.0	-	$\log_{10} k = \mathcal{N}(1.0, 0.1)$
2b	boundEGFR → EGF + UEGFR	1.0	-	$\log_{10} k = \mathcal{N}(1.0, 0.1)$
3	activeSOS → inactiveSOS	1.0	1.0	$\log_{10} k = \mathcal{N}(1.0, 0.1)$
4	inactiveRas+activeSOS → activeRas + activeSOS	1.0	1.0	$\log_{10} k = \mathcal{N}(1.0, 0.1)$
5	activeRas+Gap → inactiveRas+Gap	1.0	1.0	$\log_{10} k = \mathcal{N}(1.0, 0.1)$
6	cRaf+activeRas → cRafPP+activeRas	1.0	1.0	$\log_{10} k = \mathcal{N}(1.0, 0.1)$
7	cRafPP → cRaf	1.0	1.0	$\log_{10} k = \mathcal{N}(1.0, 0.1)$
8	MEK+cRafPP → MEKPP+cRafPP	1.0	1.0	$\log_{10} k = \mathcal{N}(1.0, 0.1)$
9	MEKPP → MEK	1.0	1.0	$\log_{10} k = \mathcal{N}(1.0, 0.1)$
10	ERK+MEKPP → ERKPP+MEKPP	1.0	1.0	$\log_{10} k = \mathcal{N}(1.0, 0.1)$
11	inactiveSOS+ERKPP → removedSOS+ERKPP	1.0	1.0	$\log_{10} k = \mathcal{N}(1.0, 0.1)$
12	activeSOS+ERKPP → removedSOS+ERKPP	1.0	1.0	$\log_{10} k = \mathcal{N}(1.0, 0.1)$
13	cRaf+PKA → removedcRaf+PKA	1.0	1.0	$\log_{10} k = \mathcal{N}(1.0, 0.1)$
14	inactivePKA+ PKAA → PKA + PKAA	1.0	1.0	$\log_{10} k = \mathcal{N}(1.0, 0.1)$
15	inactivePKA+Cilostamide → PKA + Cilostamide	1.0	1.0	$\log_{10} k = \mathcal{N}(1.0, 0.1)$
16	PKA → inactivePKA	1.0	1.0	$\log_{10} k = \mathcal{N}(1.0, 0.1)$
17	inactiveEPAC +EPACA → EPAC + EPACA	1.0	1.0	$\log_{10} k = \mathcal{N}(1.0, 0.1)$
18	inactiveEPAC +Cilostamide → EPAC + Cilostamide	1.0	1.0	$\log_{10} k = \mathcal{N}(1.0, 0.1)$
19	EPAC → inactiveEPAC	1.0	1.0	$\log_{10} k = \mathcal{N}(1.0, 0.1)$
20	inactiveRap1 + EPAC → activeRap1 + EPAC	1.0	1.0	$\log_{10} k = \mathcal{N}(1.0, 0.1)$
21	activeRap1 + Gap → inactiveRap1 + Gap	1.0	1.0	$\log_{10} k = \mathcal{N}(1.0, 0.1)$
22	BRaf + activeRap1 → BRafPP+activeRap1	1.0	1.0	$\log_{10} k = \mathcal{N}(1.0, 0.1)$
23	BRafPP → BRaf	1.0	1.0	$\log_{10} k = \mathcal{N}(1.0, 0.1)$
24	MEK + BRafPP → MEKPP + BRafPP	1.0	1.0	$\log_{10} k = \mathcal{N}(1.0, 0.1)$
25	inactiveC3G+boundEGFR → activeC3G + boundEGFR	1.0	1.0	$\log_{10} k = \mathcal{N}(1.0, 0.1)$
26	activeC3G → inactiveC3G	1.0	-	$\log_{10} k = \mathcal{N}(1.0, 0.1)$
27	inactiveRap1+activeC3G → activeRap1+activeC3G	1.0	1.0	$\log_{10} k = \mathcal{N}(1.0, 0.1)$
28	BRaf+activeRas → BRafPP+activeRas	1.0	1.0	$\log_{10} k = \mathcal{N}(1.0, 0.1)$
29	ERKPP → ERK	1.0	1.0	$\log_{10} k = \mathcal{N}(1.0, 0.1)$
30	boundEGFR → degradedEGFR	1.0	-	$\log_{10} k = \mathcal{N}(1.0, 0.1)$

[a] logarithm (base rate constant value)
[b] Base value of Michaelis constant

Table 5.4: Proposed reactions for Example 2 and 3

where

$$\zeta_j = \begin{cases} 1.0 & \text{if } j = 1 \\ 0.99 & \text{if } j = 2 \\ \zeta_{j-1}^{5/4} & \text{if } j = 3 \text{ to } 40 \end{cases} \tag{5.16}$$

The proposal for the mutation move is taken identical to the single-chain proposal. For the exchange move, we only permit exchange of samples between adjacent chains.

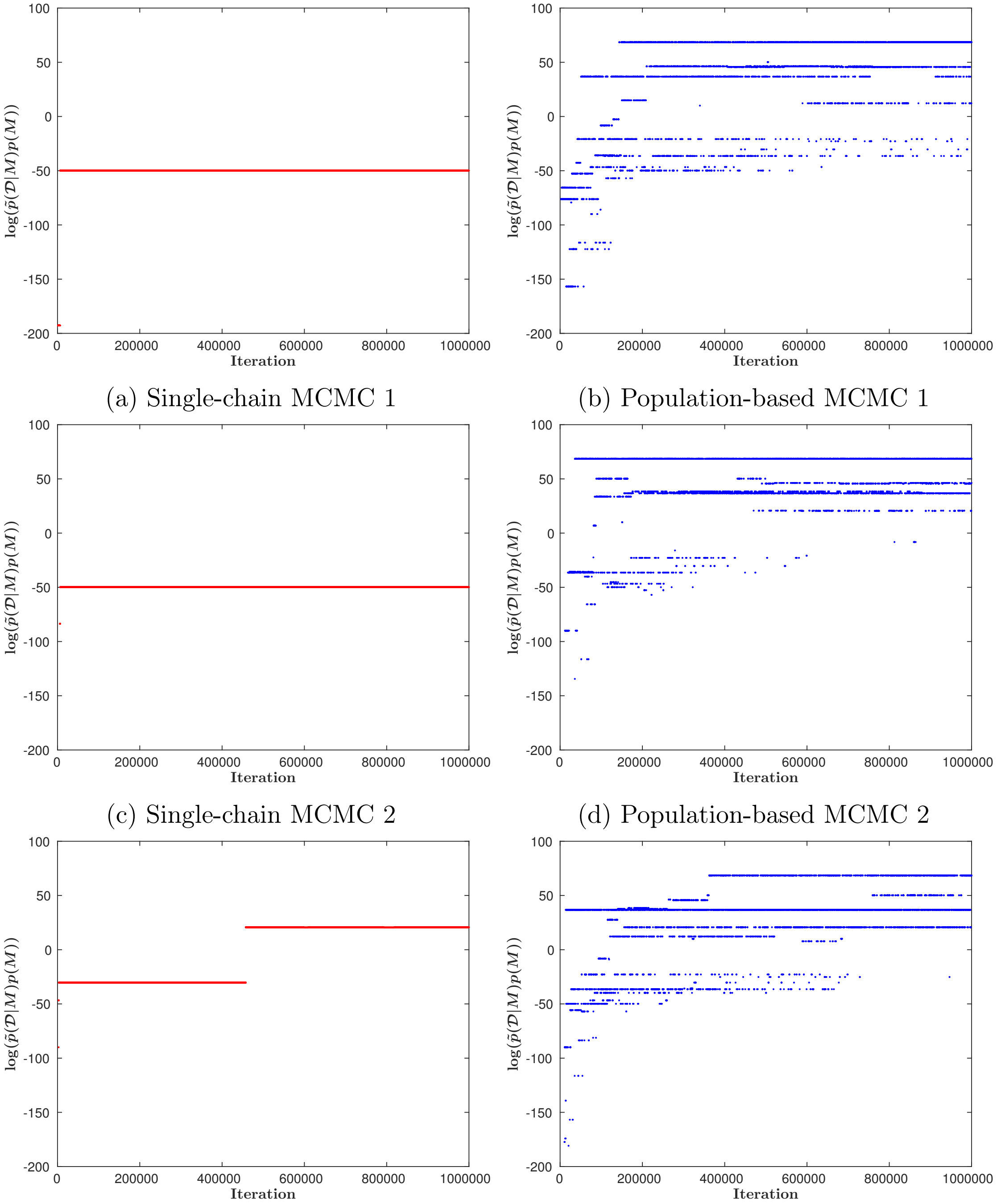
Three realizations each of the two samplers are shown in Figure 5-3. We see that with the single-chain approximate model inference algorithm, the sampler gets stuck in one of the lower posterior modes, whereas the population-based sampler explores multiple modes and quickly identifies the high-posterior modes. Thus there is a clear advantage with the population-based approximate model inference algorithm over the standard single-chain approximate model inference algorithm. In this example, we find that there are two effective networks with $\log(\tilde{p}(\boldsymbol{\mathcal{D}}|M)p(M)) \approx 68$. All others effective networks have $\log(\tilde{p}(\boldsymbol{\mathcal{D}}|M)p(M)) < 50$.

5.4.2 Example 3: 30 dimensional reaction network with multiple species observables

As our second example, we consider once again the set of proposed reactions used in Example 2, but now with multiple observables: ERK and inactiveSOS concentrations. We generate 30 i.i.d. data points each of ERK and inactiveSOS measurements with the noise model $\mathcal{N}(0, 4)$ and all rate constants and Michaelis constants set to their base values (Table 5.4). We impose independent Gaussian priors on the logarithm of rate constants, with means and variances as shown in Table 5.4. The prior probability distribution over all plausible models is taken to be uniform. The noise variance for the likelihood function, as in the data generating process, is taken as $\sigma^2 = 4$.

We generate 4 million samples using the single-chain approximate model inference and the population-based approximate model inference algorithms. We choose the proposal distribution for the single-chain sampler to be a Poisson distribution Pois(1.0) for the number of reactions r to be added/deleted and uniformly select the r reactions to added/deleted. In case of the population-based algorithm, we take a population of 40 chains and the sequence of distributions is the same as in Example 2. The proposal for the mutation move is taken identical to the single-chain proposal. For the exchange move, we only permit exchange of samples between adjacent chains.

Three realizations each of the two samplers are shown in Figure 5-5. The population-based model inference samplers can be seen to have converged. We see that with the single-chain approximate model inference algorithm, the sampler gets stuck in one of the low posterior modes, whereas the population-based sampler explores multiple modes and quickly identifies the high-posterior mode to which it converges. Thus



(a) Single-chain MCMC 1

(b) Population-based MCMC 1

(c) Single-chain MCMC 2

(d) Population-based MCMC 2

(e) Single-chain MCMC 3

(f) Population-based MCMC 3

Figure 5-3: Three realizations of single-chain approximate model inference MCMC and population-based approximate model inference MCMC algorithms for Example 2. For the population-based algorithm, we are showing the posterior samples for the chain corresponding to the target distribution $\tilde{p}(M|\mathcal{D})$

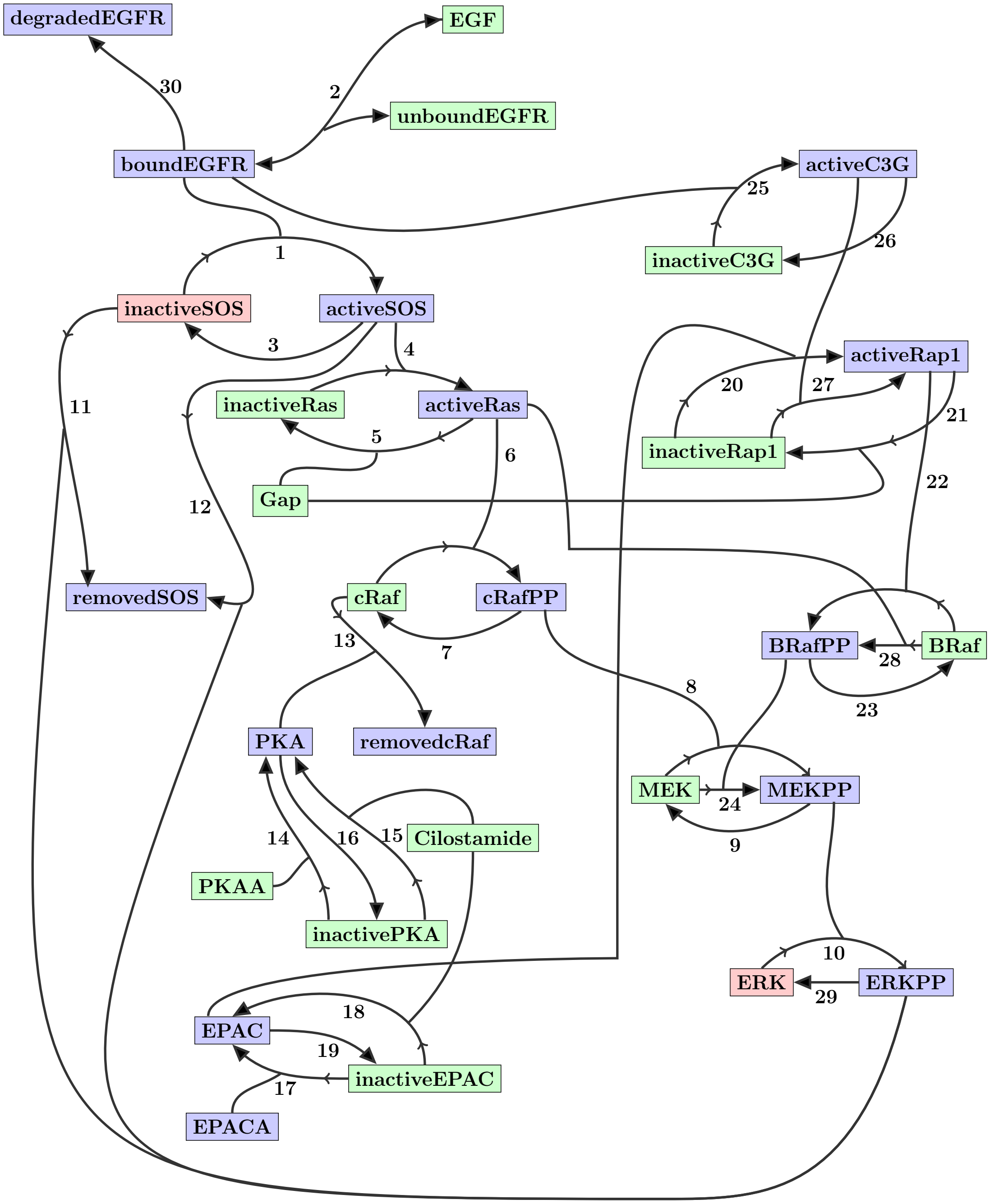
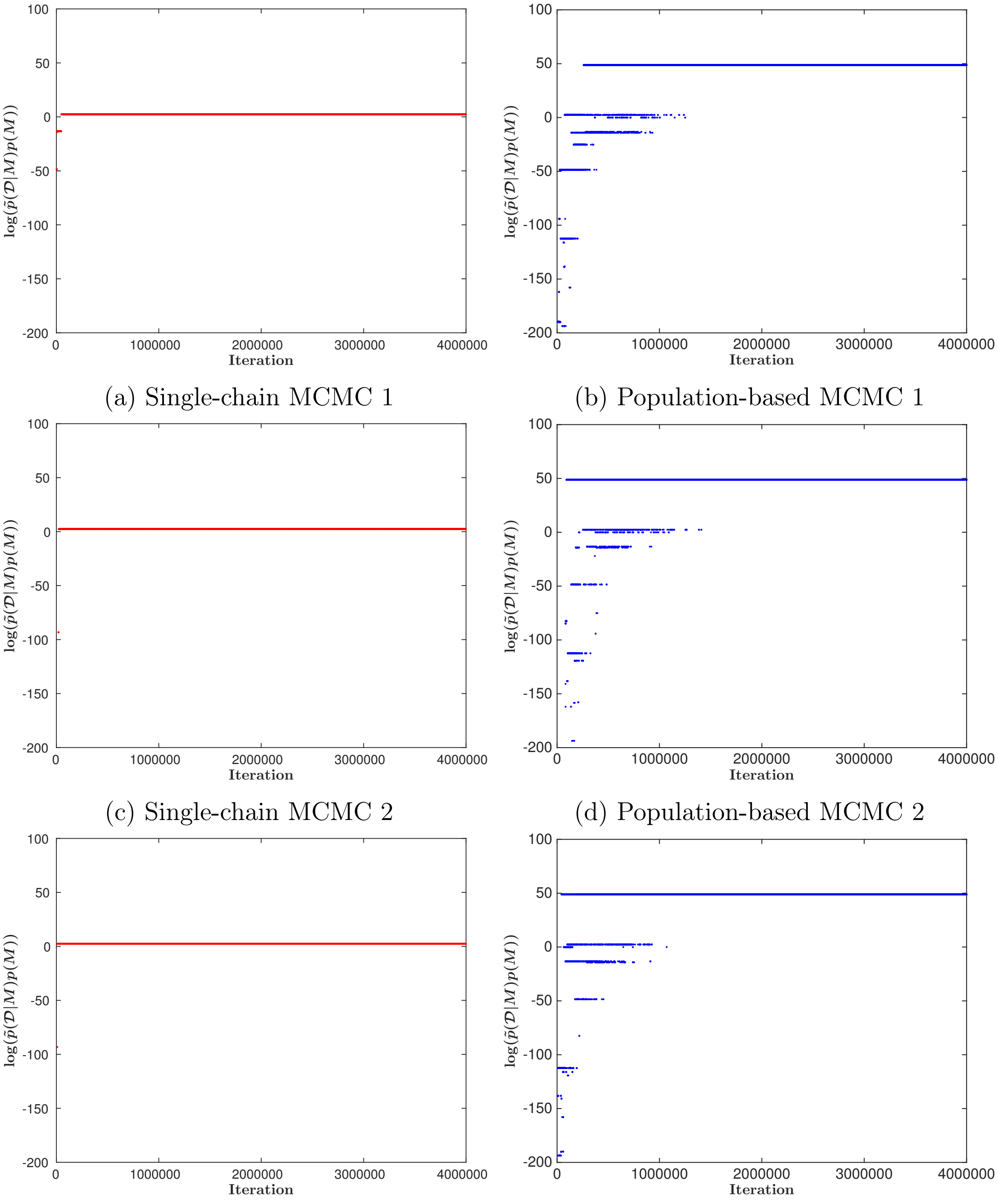


Figure 5-4: Reaction network of Example 3



(a) Single-chain MCMC 1

(b) Population-based MCMC 1

(c) Single-chain MCMC 2

(d) Population-based MCMC 2

(e) Single-chain MCMC 3

(f) Population-based MCMC 3

Figure 5-5: Three realizations of single-chain approximate model inference MCMC and population-based approximate model inference MCMC algorithms for Example 3. For the population-based algorithm, we are showing the posterior samples for the chain corresponding to the target distribution $\tilde{p}(M|\mathcal{D})$

once again we find that there is a clear advantage with the population-based approximate model inference algorithm over the standard single-chain approximate model inference algorithm. For the population-based scheme, the total number of models visited is only 160000. With ERK and inactiveSOS as the observables, the number of effective networks visited was 12000. By employing a sampling-based approach, we have therefore managed to characterize the posterior distribution over 10^9 models by evaluating the evidence of only 12000 models. In contrast, the brute-force approach of evaluating the evidence of all models is infeasible. In general, the combination of approximation of model evidence with Laplace's method and a population-based sampling method can explore very large model spaces that is infeasible by exhaustive evaluation of all evidences and very expensive with exact sampling methods.

Chapter 6

Conclusions and future work

This thesis focuses on the development of tractable numerical methods for Bayesian inference of nonlinear chemical reaction networks. Inference of reaction network models is important in areas such as biology, combustion, and catalysis. The traditional approach to inference of model structure in the Bayesian paradigm relies on evaluating a multidimensional integral for every plausible model. When the number of models is large, however, this approach is infeasible. The inference of chemical reaction networks can require the consideration of large model spaces since the number of plausible models grows combinatorially with the number of proposed reactions. Across-model samplers that jointly explore the parameter spaces of all models can enable large-scale network inference. However, the effective use of existing across-model samplers continues to be a challenge. In this thesis, we develop efficient across-model samplers for large-scale inference of reaction networks.

6.1 Conclusions

In the third chapter, we presented a fixed-dimensional interpretation of the network inference problem. Recognizing that the set of possible models in the network inference problem are nested allows us to tackle the network inference problem as a fixed-dimensional inference problem, and to adapt existing fixed-dimensional adaptive MCMC for efficient network inference. Adaptive methods allow automatic tuning of proposals in contrast to the conventional approach of manual tuning of between-model parameter proposals by pilot simulations. We employed our methodology to infer reforming models of methane on rhodium with real and synthetic data. Our examples indicate that efficient large-scale nonlinear network inference is feasible with adaptive methods even in settings where little is known about the parameter posteriors of the proposed models. Our general fixed-dimensional adaptive MCMC framework for network inference provides myriad information, including the most probable models, full uncertainty in parameter values, and the dominant pathways.

The network-based interaction of species in reaction networks translates into clusters of networks such that networks belonging to a cluster all have identical effective networks. We show in Chapter 4 how determining the *effective networks* of proposed networks in an across-model sampler can enable the construction of more effective between-model parameter proposals. The nonlinearity of chemical reaction networks means that keeping the rate constants of reactions that are commmon to the current and the proposed networks in across-model samplers can often produce poor parameter proposals. To develop better proposals in such cases, we have further developed sensitivity-based network-aware proposals that identify critical reactions and include a network-aware parameter proposal for their rate constants when proposing moves between networks. The developed methods when tested on a range of example problems show clear advantage in exploiting network topology and sensitivity of observables to network elements while designing between-model parameter proposals.

Finally, we have developed an approach for very large-scale model inference—settings when the number of models is so large that exact sampling methods are prohibitively expensive. By incorporating a Laplace approximation for model evidence, we run an MCMC simulation only over model indicators. Approximating the evidence of all models visited by the sampler eliminates the need for developing effective parameter proposals for between-model moves and thus reduces the overall cost of the simulation. The conventional single-chain Markov chain Monte Carlo simulations over model indicators still find it difficult to explore model posterior distributions, which are typically multimodal. For this, we have developed a population-based approximate model inference algorithm that involves running a population of MCMC chains which exchange information among themselves to explore the posterior distribution over models. We applied our algorithm to network inference problems from biology with spaces of around 10^9 networks. The population-based approximate model inference approach is seen to outperform the single-chain approximate model inference algorithm.

Large-scale nonlinear network inference is an important goal that could help improve our fundamental understanding of many processes and enable better predictions with quantified uncertainties. To this end, the development of efficient across-model samplers is critical. Effective across-model samplers would allow a systematic comparison of all plausible networks in contrast to the conventional approach of comparing a few hand-crafted models or using simplified linear models. This thesis provides some ideas on efficient across-models samplers for nonlinear network inference.

6.2 Future work

Our work in this thesis suggests some areas of future work for further development of network inference algorithms. We outline a few ideas here:

1. **Exact approximations to model evidences**: A key challenge to efficient across-model sampling that has been addressed in this thesis is the design of good between-model parameter proposals. The ideas presented in Chapter 3 and 4 are definite improvements over existing methods, but further advancements may be possible. One in particular is to consider using an *exact approximations* of model evidence when proposing moves between models. A recent algorith-

mic development in the MCMC literature called pseudomarginal MCMC is one such approach [6, 11]. The idea behind these exact approximations is to use an ensemble of samples to approximate the evidence of the model to which a move is proposed. As a result, the alignment of densities, which is key to good sampling performance of the methods discussed in this thesis, is circumvented. Note, in contrast to approximation based methods discussed in Chapter 5 of this thesis, the exact approximation Monte Carlo converges asymptotically to the true posterior distribution. How a good exact approximation may be constructed is a subject of ongoing research, but expoiting network-based species interactions as in Chapter 4 is something that can be immediately incorporated for network inference problems.

2. **Hybrid across-model samplers**: We presented adaptive MCMC methods for network inference in Chapter 3. It is well known, however, that the performance of adaptive MCMC methods for finite samples depends on initially having a nonadaptive MCMC algorithm with a fairly good sampling performance. To this end, we have developed methods that improve the performace of between-model proposals. One interesting idea could be to combine the ideas from Chapter 3 and 4 by using network-aware parameter proposals and adapting the model-move proposals $q(M'|M)$ based on previous posterior samples.

3. **Network-aware forward models**: The exclusion of a reaction from a network has been implemented in this thesis by setting the corresponding rate constant to zero. As a result, the forward model solve still has a computational complexity $O(N_s^\alpha)$, where N_s is the total number of species in a network with all proposed reactions, even when the actual number of species in the network is much smaller. For large problems, superior computational performance may be derived by turning the forward model also network aware, i.e., the forward model need only perform computations on the species that are part of the cur-

rent network.

4. **Limiting the size of between-model moves**: The network-aware methods of Chapter 4 design between-model parameter proposals based on the current and proposed effective networks. Currently there is no limit on the size of allowed difference between the effective networks. For large network inference problem, proposed moves may often be between effective networks that differ greatly in their sizes. As has been discussed in this thesis, in general, it is hard to construct efficient proposals in high dimensions. Therefore, one idea that can be incorporated in across-model samplers for network inference is to limit the difference in sizes of the effective networks between which proposals are made.

5. **Approximation-based methods with better asymptotic performance**: The Laplace's method discussed in this thesis has an $O(n_{data}^{-1})$ approximation error. Future work could look at approximation schemes that have faster asymptotic convergence rates. This may require considering higher-order Taylor expansions and/or incorporating more information about the structure of the network inference forward models, priors, etc, into in the evidence approximation.

Appendix A

Online expectation-maximization for proposal adaptation

Here we present a derivation of the online EM algorithm applied to a general point mass mixture proposal:

$$q(k_i;\boldsymbol{\psi_i}) = b_{i,0}\delta(k_i) + \sum_{m=1}^{M} b_{i,m} q_m(k_i;\theta_{i,m}). \tag{A.1}$$

The marginal proposal distribution $q(k_i;\boldsymbol{\psi_i})$ shown in (A.1) can also be rewritten as

$$q(k_i;\boldsymbol{\psi_i}) = \sum_{z_i} q(k_i, z_i;\boldsymbol{\theta_i}), \tag{A.2}$$

and taking $q(k_i;\boldsymbol{\psi_i})$ to be independent for each k_i, the joint proposal distribution for an N-dimensional problem follows:

$$q(\bar{\boldsymbol{k}};\bar{\boldsymbol{\psi}}) = \sum_{\bar{\boldsymbol{z}}} q(\bar{\boldsymbol{k}},\bar{\boldsymbol{z}};\bar{\boldsymbol{\psi}}) = \prod_{i=1}^{N}\sum_{z_i} q(k_i, z_i;\boldsymbol{\theta_i}). \tag{A.3}$$

Here, z_i is a *latent variable* that takes one of $M+1$ values corresponding to the $M+1$ components that could generate the posterior sample. $q(\bar{\boldsymbol{k}},\bar{\boldsymbol{z}})$ is the joint distribution of $\bar{\boldsymbol{z}}$ and $\bar{\boldsymbol{k}}$ and is referred to as the complete-data likelihood. Expanding (A.2) by the product rule of probability gives:

$$\begin{aligned} q(k_i; \boldsymbol{\psi_i}) &= \sum_{z_i} q(k_i|z_i) q(z_i) \\ &= q(z_i = 0)\delta(k_i) + \sum_{m=1}^{M} q(z_i = m) q_m(k_i|z_i = m; \theta_{i,m}); \end{aligned} \tag{A.4}$$

Comparing (A.4) to (A.1), we see that

$$q(z_i = 0) = b_{i,0} \quad \text{and} \quad q(z_i = m) = b_{i,m}. \tag{A.5}$$

After the steps for a general point mass mixture proposal have been established, we will obtain specific expressions for the case when the continuous components of the above proposal distribution (3.9) are all Gaussian.

A.0.1 KL divergence minimization yields a maximum likelihood problem

Recall that our goal is to update the proposal distribution $q(\bar{\boldsymbol{k}}; \bar{\boldsymbol{\psi}})$ iteratively based on samples from the posterior distribution $p(\bar{\boldsymbol{k}}|\boldsymbol{\mathcal{D}})$ so as to minimize the KL divergence:

$$D_{KL}(p(\bar{\boldsymbol{k}}|\boldsymbol{\mathcal{D}} \| q(\bar{\boldsymbol{k}}; \bar{\boldsymbol{\psi}})) = \int p(\bar{\boldsymbol{k}}|\boldsymbol{\mathcal{D}}) \log\left(\frac{p(\bar{\boldsymbol{k}}|\boldsymbol{\mathcal{D}})}{q(\bar{\boldsymbol{k}}; \bar{\boldsymbol{\psi}})}\right) d\bar{\boldsymbol{k}} \tag{A.6}$$

w.r.t. the proposal parameters $\bar{\boldsymbol{\psi}}$. Note that minimizing the KL divergence in (A.6) is equivalent to maximizing the cross entropy $\int p(\bar{\boldsymbol{k}}|\boldsymbol{\mathcal{D}}) q(\bar{\boldsymbol{k}}, \bar{\boldsymbol{\psi}}) d\bar{\boldsymbol{k}}$. Thus the objective function can be rewritten as

$$\bar{\boldsymbol{\psi}}^* = \arg\max_{\bar{\boldsymbol{\psi}}} \int p(\bar{\boldsymbol{k}}|\boldsymbol{\mathcal{D}}) \log(q(\bar{\boldsymbol{k}}; \bar{\boldsymbol{\psi}})) d\bar{\boldsymbol{k}}. \tag{A.7}$$

The integral in (A.7) can be approximated by a Monte Carlo sum using T samples from the posterior distribution $p(\bar{\boldsymbol{k}}|\boldsymbol{\mathcal{D}})$ as

$$I = \frac{1}{T}\sum_{t=1}^{T}\log(q(\bar{\boldsymbol{k}}^{\boldsymbol{t}};\bar{\boldsymbol{\psi}})) = \frac{1}{T}\log\left(\prod_{t=1}^{T} q(\bar{\boldsymbol{k}}^{\boldsymbol{t}};\bar{\boldsymbol{\psi}})\right). \tag{A.8}$$

Now, if we think of $\bar{\boldsymbol{k}}^{\boldsymbol{t=1:T}}$ as pseudo-data and $q(\bar{\boldsymbol{k}}^{\boldsymbol{t}};\boldsymbol{\theta})$ as a likelihood, cross entropy can be interpreted as a log-likelihood under infinite data and (A.7) as a maximum (log-)likelihood problem. Mathematically (A.7) can also be written as

$$\bar{\boldsymbol{\psi}}^{\boldsymbol{*}} = \arg\max_{\bar{\boldsymbol{\psi}}}\ \lim_{T\to\infty}\frac{1}{T}\log\left(\prod_{t=1}^{T} q(\bar{\boldsymbol{k}}^{\boldsymbol{t}};\bar{\boldsymbol{\psi}})\right) \tag{A.9}$$

A.0.2 Classical EM algorithm

Suppose we are given T independent samples $(\bar{\boldsymbol{k}}^{\boldsymbol{1}},\bar{\boldsymbol{k}}^{\boldsymbol{2}},\ldots,\bar{\boldsymbol{k}}^{\boldsymbol{T}})$ distributed according to $p(\bar{\boldsymbol{k}}|\boldsymbol{\mathcal{D}})$. The solution of the maximum log-likelihood problem

$$\bar{\boldsymbol{\psi}}^{\boldsymbol{*}} = \arg\max_{\bar{\boldsymbol{\psi}}}\frac{1}{T}\sum_{t=1}^{T}\log\left(q(\bar{\boldsymbol{k}}^{\boldsymbol{t}};\bar{\boldsymbol{\psi}})\right) \tag{A.10}$$

can be obtained by taking the derivative of the log-likelihood and solving the resulting nonlinear equations. The nonlinear equations thus obtained seldom have a closed-form solution and thus are solved by numerical optimization.

An alternative known as expectation-maximization algorithm exists for the solution of the maximum log-likelihood problem [31, 15]. The EM algorithm often results in simple analytical expressions and avoids the difficulties of gradient-based optimization approaches. The EM algorithm consists of two steps, known as the E-step and M-step, that are solved iteratively to obtain the optimal parameter values under mild regularity conditions [117]. The two steps are given by

E-step

$$
\begin{aligned}
Q(\bar{\boldsymbol{\psi}}, \bar{\boldsymbol{\psi}}_{\boldsymbol{n-1}}) &= \int \log\left(\prod_{t=1}^{T} q(\bar{\boldsymbol{k}}^{\boldsymbol{t}}, \bar{\boldsymbol{z}}^{\boldsymbol{t}}; \bar{\boldsymbol{\psi}})\right)\left(\prod_{t=1}^{T} q(\bar{\boldsymbol{z}}^{\boldsymbol{t}}|\bar{\boldsymbol{k}}^{\boldsymbol{t}}, \bar{\boldsymbol{\psi}}_{\boldsymbol{n-1}})\right) d\bar{\boldsymbol{z}}^{\boldsymbol{1}} \dots d\bar{\boldsymbol{z}}^{\boldsymbol{T}} \\
&= \mathbb{E}_{\bar{\boldsymbol{z}}^{\boldsymbol{1}} \dots \bar{\boldsymbol{z}}^{\boldsymbol{T}}}\left[\log\left(\prod_{t=1}^{T} q(\bar{\boldsymbol{k}}^{\boldsymbol{t}}, \bar{\boldsymbol{z}}^{\boldsymbol{t}}; \bar{\boldsymbol{\psi}})\right)\right] \qquad \text{(A.11)}
\end{aligned}
$$

M-step

$$
\bar{\boldsymbol{\psi}}_{\boldsymbol{n}} = \arg\max_{\bar{\boldsymbol{\psi}}} Q(\bar{\boldsymbol{\psi}}, \bar{\boldsymbol{\psi}}_{\boldsymbol{n-1}}) \qquad \text{(A.12)}
$$

The E-step in the above equations evaluates the expectation of the logarithm of the complete-data likelihood, where the expectation is taken with respect to the latent variables conditioned on available (sampled) rate parameters. In the M-step, an updated set of parameter values are computed by maximizing the expected log-likelihood from the E-step. The EM algorithm as described in (A.11) and (A.12) is applicable if all the observed samples ($\bar{\boldsymbol{k}}^{\boldsymbol{t=1:T}}$) are available a priori and the samples are independent.

Our problem is different from the above case since we are generating samples from $p(\bar{\boldsymbol{k}}|\boldsymbol{\mathcal{D}})$ in batches. Moreover, the generated samples are not independent as they are coming from an MCMC scheme. Thus we use a sequential variant of the EM algorithm known as the online EM algorithm and specify conditions under which the resulting adaptive MCMC algorithm converges to the posterior distribution, $p(\bar{\boldsymbol{k}}|\boldsymbol{\mathcal{D}})$.

A.0.3 Online expectation maximization

We begin our discussion of the online EM algorithm by assuming that the proposal distribution $q(\bar{\boldsymbol{k}}, \bar{\boldsymbol{z}}; \bar{\boldsymbol{\psi}})$ can be represented in the form

$$
q(\bar{\boldsymbol{k}}, \bar{\boldsymbol{z}}; \bar{\boldsymbol{\psi}}) = \exp(\langle \boldsymbol{s}(\bar{\boldsymbol{k}}, \bar{\boldsymbol{z}}), \bar{\boldsymbol{\phi}}(\bar{\boldsymbol{\psi}})\rangle - \bar{\boldsymbol{A}}(\bar{\boldsymbol{\psi}})). \qquad \text{(A.13)}
$$

Distributions that can be cast in the above form are known to belong to the exponential family [13]. Here, $\boldsymbol{s}(\bar{\boldsymbol{k}}, \bar{\boldsymbol{z}})$ is a vector of sufficient statistics, $\bar{\boldsymbol{\phi}}(\bar{\boldsymbol{\psi}})$ refers to the natural parameters, and $\bar{\boldsymbol{A}}(\bar{\boldsymbol{\psi}})$ is the log base distribution. The operator $\langle\cdot\rangle$ is the standard inner product. Plugging the above expression for $q(\bar{\boldsymbol{k}}, \bar{\boldsymbol{z}})$ into (A.11) and (A.12), we get

E-step:

$$Q(\bar{\boldsymbol{\psi}}, \bar{\boldsymbol{\psi}}_{\boldsymbol{n-1}}) = \int \prod_{t=1}^{T} q(\bar{\boldsymbol{z}}^{\boldsymbol{t}} | \bar{\boldsymbol{k}}^{\boldsymbol{t}}; \bar{\boldsymbol{\psi}}_{\boldsymbol{n-1}}) \sum_{t=1}^{T} \left(\langle \boldsymbol{s}(\bar{\boldsymbol{k}}^{\boldsymbol{t}}, \bar{\boldsymbol{z}}^{\boldsymbol{t}}), \bar{\boldsymbol{\phi}}(\bar{\boldsymbol{\psi}}) \rangle - \bar{\boldsymbol{A}}(\bar{\boldsymbol{\psi}}) \right) d\bar{\boldsymbol{z}}^{\boldsymbol{1}} \dots d\bar{\boldsymbol{z}}^{\boldsymbol{T}}, \tag{A.14}$$

M-step:

$$\bar{\boldsymbol{\psi}}_{\boldsymbol{n}} = \arg\max_{\bar{\boldsymbol{\psi}}} Q(\bar{\boldsymbol{\psi}}, \bar{\boldsymbol{\psi}}_{\boldsymbol{n-1}}). \tag{A.15}$$

The above expectation and maximization steps can be recast in terms of sufficient statistics as

E-step:

$$\boldsymbol{S}_{\boldsymbol{n}}^{\boldsymbol{T}} = \frac{1}{T} \sum_{t=1}^{T} \mathbb{E}_{\bar{\psi}_{\boldsymbol{n-1}}^{\boldsymbol{T}}} \left[\boldsymbol{s}(\bar{\boldsymbol{k}}^{t}, \bar{\boldsymbol{z}}^{t}) | \bar{\boldsymbol{k}}^{t} \right], \tag{A.16}$$

M-step:

$$\bar{\boldsymbol{\psi}}_{\boldsymbol{n}}^{\boldsymbol{T}} = \Gamma\{\boldsymbol{S}_{\boldsymbol{n}}^{\boldsymbol{T}}\}, \tag{A.17}$$

where $\Gamma\{\boldsymbol{S}_{\boldsymbol{n}}^{\boldsymbol{T}}\} = \arg\max_{\bar{\boldsymbol{\psi}}} (\langle \boldsymbol{S}_{\boldsymbol{n}}^{\boldsymbol{T}}, \bar{\boldsymbol{\phi}}(\bar{\boldsymbol{\psi}}) \rangle - \bar{\boldsymbol{A}}(\bar{\boldsymbol{\psi}}))$. Letting $T \to \infty$, the EM iterations are

E-step:

$$\boldsymbol{S}_{\boldsymbol{n}} = \mathbb{E}_{p(\bar{\boldsymbol{k}}|\boldsymbol{\mathcal{D}})} \left(\mathbb{E}_{\bar{\boldsymbol{\psi}}_{\boldsymbol{n-1}}} \left[\boldsymbol{s}(\bar{\boldsymbol{k}}, \bar{\boldsymbol{z}} | \bar{\boldsymbol{k}}) \right] \right) \tag{A.18}$$

M-step:

$$\bar{\boldsymbol{\psi}}_{\boldsymbol{n}} = \Gamma\{\boldsymbol{S}_{\boldsymbol{n}}\} \tag{A.19}$$

Thus our overall goal of solving (A.7) is equivalent to locating the solutions of

$$\mathbb{E}_{p(\bar{\boldsymbol{k}}|\boldsymbol{\mathcal{D}})}\left(\mathbb{E}_{\Gamma\{\boldsymbol{S}\}}\left[\boldsymbol{s}(\bar{\boldsymbol{k}},\bar{\boldsymbol{z}}|\bar{\boldsymbol{k}})\right]\right)-\boldsymbol{S}=0 \tag{A.20}$$

If we now take $\frac{1}{T}\sum_{t=1}^{T}\mathbb{E}_{\Gamma\{\boldsymbol{S}\}}[\boldsymbol{s}(\bar{\boldsymbol{k}}^{\boldsymbol{t}},\bar{\boldsymbol{z}}^{\boldsymbol{t}})|\bar{\boldsymbol{k}}^{\boldsymbol{t}}]$ to be a noisy estimate of $\mathbb{E}_{p(\bar{\boldsymbol{k}}|\boldsymbol{\mathcal{D}})}\left(\mathbb{E}_{\Gamma\{\boldsymbol{S}\}}\left[\boldsymbol{s}(\bar{\boldsymbol{k}},\bar{\boldsymbol{z}}|\bar{\boldsymbol{k}})\right]\right)$, application of the Robbins-Monro stochastic approximation algorithm results in the online EM algorithm [4, 104]. The online EM iterations are given by

E-step:

$$\boldsymbol{S_n}=(1-\eta_n)\boldsymbol{S_{n-1}}+\eta_n\left(\frac{1}{T}\sum_{t=1}^{T}\mathbb{E}_{\bar{\psi}_{\boldsymbol{n-1}}}[\boldsymbol{s}(\bar{\boldsymbol{k}}_{\boldsymbol{t}},\bar{\boldsymbol{z}}_{\boldsymbol{t}})|\bar{\boldsymbol{k}}_{\boldsymbol{t}}]\right) \tag{A.21}$$

M-step:

$$\bar{\boldsymbol{\psi}}_{\boldsymbol{n}}=\Gamma\{\boldsymbol{S_n}\} \tag{A.22}$$

η_n here is a sequence of decreasing positive step sizes and satisfies the following two conditions:

$$\sum_{n=1}^{\infty}\eta_n=\infty \text{ and } \sum_{n=1}^{\infty}\eta_n^2<\infty \tag{A.23}$$

We take $\eta_n=1/n$ in our work. We now return to the complete-data likelihood of the point-mass mixture proposal distribution ((A.2, A.3)). Assuming that the continuous parts of the proposal distribution for each rate parameter k_i are Gaussian distributions with arbitrary initial means and variances and recalling that the proposal for each k_i is independent, we obtain the complete-data log-likelihood as

$$\log q(\bar{\boldsymbol{k}},\bar{\boldsymbol{z}}|\bar{\boldsymbol{\psi}})=\sum_{i=1}^{N}\sum_{m=0}^{M}z_{i,m}\log b_{i,m}+\sum_{i=1}^{N}\sum_{m=1}^{M}z_{i,m}\log\mathcal{N}(k_i;\mu_{i,m},\sigma^2_{i,m}). \tag{A.24}$$

It can be easily be shown that (A.24) can be cast in the form of (A.13) and that the corresponding sufficient statistics are given by:

For i = 1 to N

For $m = 0$ to M :

$$O_{i,m} = \frac{1}{T}\sum_{t=1}^{T}\gamma(z_{i,m}^t),$$

For $m = 1$ to M :

$$P_{i,m} = \frac{1}{T}\sum_{\substack{t=1\\k_i^t\neq 0}}^{T}\gamma(z_{i,m}^t) \qquad Q_{i,m} = \frac{1}{T}\sum_{\substack{t=1\\k_i^t\neq 0}}^{T}\gamma(z_{i,m}^t)k_i^t \qquad R_{i,m} = \frac{1}{T}\sum_{\substack{t=1\\k_i^t\neq 0}}^{T}\gamma(z_{i,m}^t)(k_i^t)^2$$

where $\gamma(z_{i,m}^t) = p(z_{i,m}^t|k_i^t;\psi_i)$ is given by

$$\gamma(z_{i,m}^t) = \begin{cases} 1 & \text{if } k_i^t = 0 \text{ and } m = 0 \\ 0 & \text{if } k_i^t = 0 \text{ and } m \neq 0 \\ 0 & \text{if } k_i^t \neq 0 \text{ and } m = 0 \\ \frac{b_{i,m}\mathcal{N}(k_i^t;\mu_{i,m},\sigma_{i,m}^2)}{\sum_{m'=1}^{M} b_{i,m'}\mathcal{N}(k_i^t;\mu_{i,m'},\sigma_{i,m'}^2)} & \text{if } k_i^t \neq 0 \text{ and } m \neq 0. \end{cases} \tag{A.25}$$

Thus the online EM iterations consist of the following two steps

E-step:

$$\begin{aligned} S_n^{O_{i,m}} &= S_{n-1}^{O_{i,m}} + \eta_n(O_{i,m} - S_{n-1}^{O_{i,m}}) \\ S_n^{P_{i,m}} &= S_{n-1}^{P_{i,m}} + \eta_n(P_{i,m} - S_{n-1}^{P_{i,m}}) \\ S_n^{Q_{i,m}} &= S_{n-1}^{Q_{i,m}} + \eta_n(Q_{i,m} - S_{n-1}^{Q_{i,m}}) \\ S_n^{R_{i,m}} &= S_{n-1}^{R_{i,m}} + \eta_n(R_{i,m} - S_{n-1}^{R_{i,m}}) \end{aligned} \tag{A.26}$$

M-step:

$$
\begin{aligned}
b_{i,m} &= \frac{S_n^{O_{i,m}}}{\sum_{m'=0}^{M} S_n^{O_{i,m'}}} \\
\mu_{i,m} &= \frac{S_n^{Q_{i,m}}}{S_n^{P_{i,m}}} \\
\sigma_{i,m}^2 &= \frac{\mu_{i,m}^2 S_n^{P_{i,m}} - 2\mu_{i,m} S_n^{Q_{i,m}} + S_n^{R_{i,m}}}{S_n^{P_{i,m}}}
\end{aligned} \tag{A.27}
$$

We have thus arrived at the steps of an adaptive MCMC algorithm that involves simulating a batch of T samples from the posterior distribution in each iteration and updating the proposal parameters based on (A.26) and (A.27). Because online EM adjusts proposal parameters based on all the past samples, standard proofs that guarantee asymptotic convergence of non-adaptive MCMC methods do not apply here. [4] provide rigorous technical conditions that guarantee a law of large numbers and a central limit theorem for the online EM algorithm. These conditions also require that one include a non-adaptive fixed component in the proposal distribution; we do so in our simulations in the form of a multi-dimensional Gaussian with fixed parameters $\tilde{q}(\bar{\boldsymbol{k}}; \tilde{\boldsymbol{\psi}})$. [106], in contrast, develop simpler conditions that ensure convergence of the adaptive MCMC scheme to the target distribution and provide a law of large numbers. The first is known as diminishing adaptation, which requires that the magnitude of adaptation is continuously decreasing. The online EM-based adaptive MCMC approach described above satisfies this condition since the step size $\eta_n \to 0$. The second condition, known as bounded convergence, is satisfied as long as the non-adaptive component $\tilde{q}(\bar{\boldsymbol{k}}; \tilde{\boldsymbol{\psi}})$ has sufficiently heavy tails or the support of $\bar{\boldsymbol{k}}$ is compact [73].

Appendix B

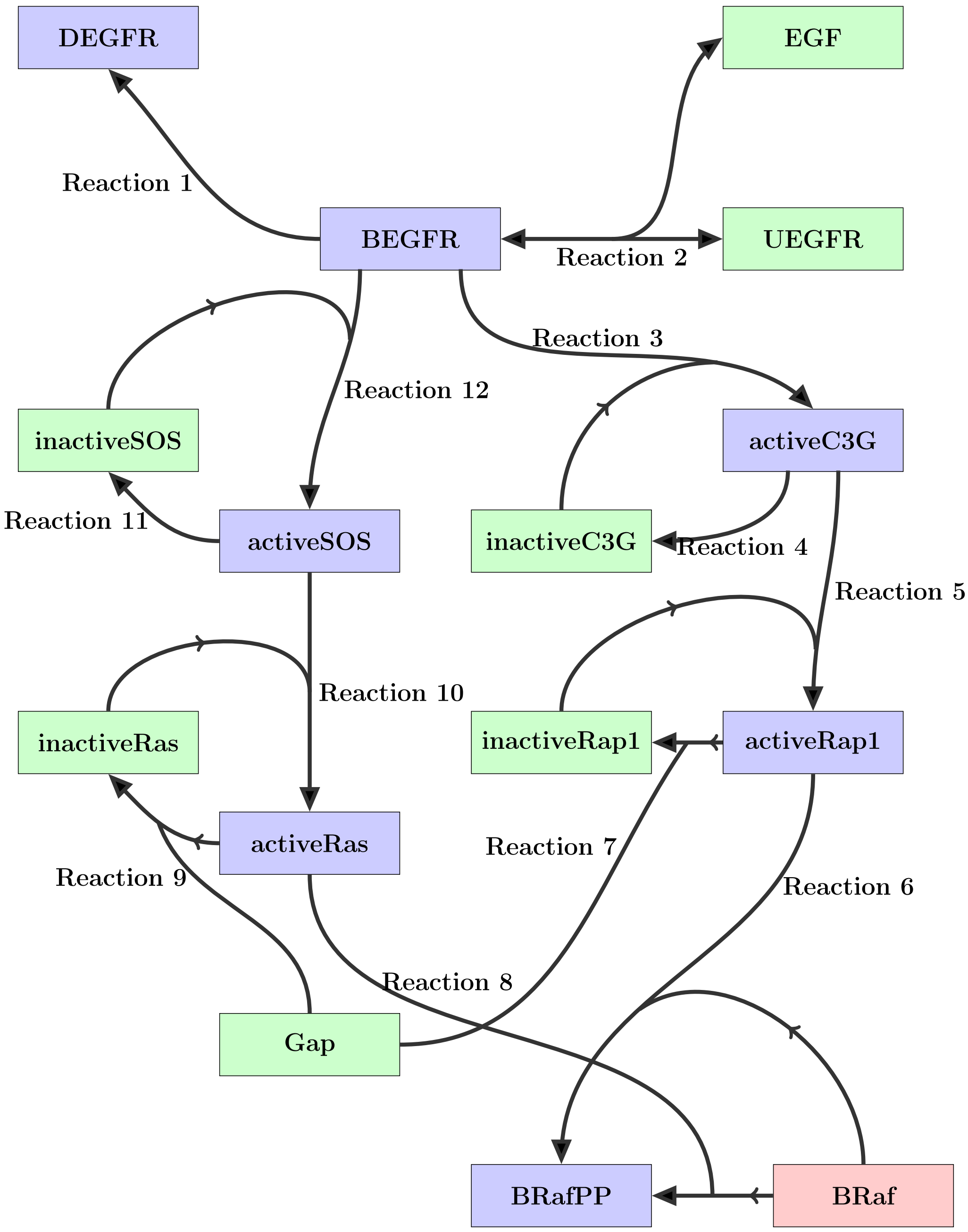
Reaction networks: reactions, reaction rates, and species production rates

B.1 12-dimensional reaction network

Here we present the details of the set of proposed reactions, the corresponding reaction and species production rate ODE expressions for the 12-reaction network used in example problems of Chapters 4 and 5 of this thesis.

B.1.1 Reactions

1. $boundEGFR \rightarrow degradedEGFR$
2. $EGF + unboundEGFR \leftrightarrow boundEGFR$
3. $inactiveC3G + boundEGFR \rightarrow activeC3G + boundEGFR$
4. $activeC3G \rightarrow inactiveC3G$
5. $inactiveRap1 + activeC3G \rightarrow activeRap1 + activeC3G$


DEGFR
EGF
Reaction 1
BEGFR
UEGFR
Reaction 2
Reaction 3
Reaction 12
inactiveSOS
activeC3G
Reaction 11
activeSOS
inactiveC3G
Reaction 4
Reaction 5
Reaction 10
inactiveRas
inactiveRap1
activeRap1
activeRas
Reaction 7
Reaction 9
Reaction 6
Reaction 8
Gap
BRafPP
BRaf

6. $BRaf + activeRap1 \rightarrow BRafPP + activeRap1$

7. $activeRap1 + Gap \rightarrow inactiveRap1 + Gap$

8. $BRaf + activeRas \rightarrow BRafPP + activeRas$

9. $activeRas + Gap \rightarrow inactiveRas + Gap$

10. $inactiveRas + activeSOS \rightarrow activeRas + activeSOS$

11. $activeSOS \rightarrow inactiveSOS$

12. $inactiveSOS + boundEGFR \rightarrow activeSOS + boundEGFR$

B.1.2 Reaction rates

1. $k_1[boundEGFR]$
2. $k_{2f}[EGF][unboundEGFR] - \ k_{2r}[boundEGFR]$
3. $\dfrac{k_3[boundEGFR][inactiveC3G]}{k_3' + [inactiveC3G]}$
4. $k_4[activeC3G]$
5. $\dfrac{k_5[activeC3G][inactiveRap1]}{k_5' + [inactiveRap1]}$
6. $\dfrac{k_6[activeRap1][BRaf]}{k_6' + [BRaf]}$
7. $\dfrac{k_7[Gap][activeRap1]}{k_7' + [activeRap1]}$
8. $\dfrac{k_8[activeRas][BRaf]}{k_8' + [BRaf]}$
9. $\dfrac{k_9[Gap][activeRas]}{k_9' + [activeRas]}$
10. $\dfrac{k_{10}[activeSOS][inactiveRas]}{k_{10}' + inactiveRas}$
11. $\dfrac{k_{11}[activeSOS]}{k_{11}' + [activeSOS]}$
12. $\dfrac{k_{12}[boundEGFR][inactiveSOS]}{k_{12}' + [inactiveSOS]}$

B.1.3 Species production rates

1. $\dot{[unboundEGFR]} = -k_{2f}[EGF][unboundEGFR] + k_{2r}[boundEGFR]$
2. $\dot{[inactiveSOS]} = -\dfrac{k_{12}[boundEGFR][inactiveSOS]}{k'_{12} + [inactiveSOS]} + \dfrac{k_{11}[activeSOS]}{k'_{11} + [activeSOS]}$
3. $\dot{[inactiveRas]} = -\dfrac{k_{10}[activeSOS][inactiveRas]}{k'_{10} + [inactiveRas]} + \dfrac{k_9[Gap][activeRas]}{k'_9 + [activeRas]}$
4. $\dot{[inactiveRap1]} = \dfrac{k_7[Gap][activeRap1]}{k'_7 + [activeRap1]} - \dfrac{k_5[activeC3G][inactiveRap1]}{k'_5 + [inactiveRap1]}$
5. $\dot{[boundEGFR]} = k_{2f}[EGF][unboundEGFR] - k_{2r}[boundEGFR] - k_1[boundEGFR]$
6. $\dot{[activeSOS]} = \dfrac{k_{12}[boundEGFR][inactiveSOS]}{k'_{12} + [inactiveSOS]} - \dfrac{k_{11}[activeSOS]}{k'_{11} + [activeSOS]}$
7. $\dot{[activeRas]} = \dfrac{k_{10}[activeSOS][inactiveRas]}{k'_{10} + [inactiveRas]} - \dfrac{k_9[Gap][activeRas]}{k'_9 + [activeRas]}$
8. $\dot{[activeRap1]} = \dfrac{k_7[Gap][activeRap1]}{k'_7 + [activeRap1]} + \dfrac{k_5[activeC3G][inactiveRap1]}{k'_5 + [inactiveRap1]}$
9. $\dot{[EGF]} = -k_{2f}[EGF][unboundEGFR] + k_{2r}[boundEGFR]$
10. $\dot{[BRafPP]} = \dfrac{k_6[activeRap1][BRaf]}{k'_6 + [BRaf]} + \dfrac{k_8[activeRas][BRaf]}{k'_8 + [BRaf]}$
11. $\dot{[BRaf]} = -\dfrac{k_6[activeRap1][BRaf]}{k'_6 + [BRaf]} - \dfrac{k_8[activeRas][BRaf]}{k'_8 + [BRaf]}$
12. $\dot{[activeC3G]} = \dfrac{k_3[boundEGFR][inactiveC3G]}{k'_3 + [inactiveC3G]} - k_4[activeC3G]$
13. $\dot{[inactiveC3G]} = -\dfrac{k_3[boundEGFR][inactiveC3G]}{k'_3 + [inactiveC3G]} + k_4[activeC3G]$
14. $\dot{[degradedEGFR]} = k_1[boundEGFR]$
15. $\dot{[Gap]} = 0$

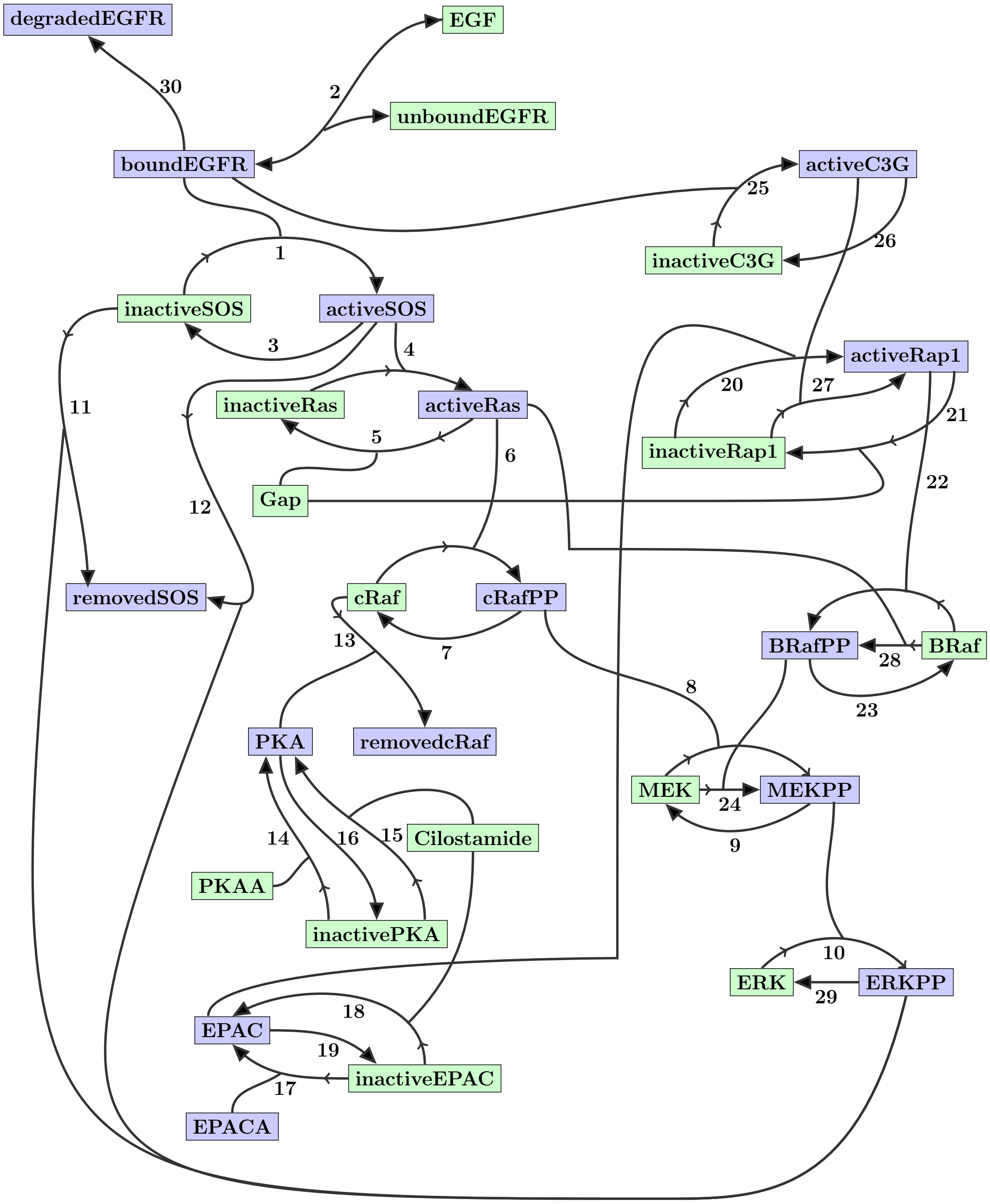
B.1.4 Initial species concentrations

All simulations using the above reactions are performed with the following initial concentrations:

1. $[unboundEGFR]_0 = 500$
2. $[inactiveSOS]_0 = 1200$
3. $[inactiveRas]_0 = 1200$
4. $[inactiveRap1]_0 = 1200$
5. $[boundEGFR]_0 = 0$
6. $[activeSOS]_0 = 0$
7. $[activeRas]_0 = 0$
8. $[activeRap1]_0 = 0$
9. $[EGF]_0 = 1000$
10. $[BRafPP]_0 = 0$
11. $[BRaf]_0 = 1500$
12. $[activeC3G]_0 = 0$
13. $[inactiveC3G]_0 = 1200$
14. $[degradedEGFR]_0 = 0$
15. $[Gap]_0 = 2400$

B.2 30-dimensional reaction network

Here we present the details of the set of proposed reactions, the corresponding reaction and species production rate ODE expressions for the 30-reaction network used in example problems of Chapter 5 of this thesis.


degradedEGFR
EGF
30
2
unboundEGFR
boundEGFR
activeC3G
25
26
1
inactiveC3G
inactiveSOS
activeSOS
3
4
activeRap1
20
27
inactiveRas
activeRas
21
11
5
inactiveRap1
6
22
Gap
12
removedSOS
cRaf
cRafPP
BRafPP
BRaf
28
13
7
23
8
PKA
removedcRaf
MEK
MEKPP
24
14
16
15
Cilostamide
9
PKAA
inactivePKA
10
ERK
ERKPP
29
18
EPAC
19
inactiveEPAC
17
EPACA

B.2.1 Reactions

1. $inactiveSOS + boundEGFR \rightarrow activeSOS + boundEGFR$

2. $EGF + unboundEGFR \leftrightarrow boundEGFR$

3. $activeSOS \rightarrow inactiveSOS$

4. $inactiveRas + activeSOS \rightarrow activeRas + activeSOS$

5. $activeRas + Gap \rightarrow inactiveRas + Gap$

6. $cRaf + activeRas \rightarrow cRafPP + activeRas$

7. $cRafPP \rightarrow cRaf$

8. $MEK + cRafPP \rightarrow MEKPP + cRafPP$

9. $MEKPP \rightarrow MEK$

10. $ERK + MEKPP \rightarrow ERKPP + MEKPP$

11. $inactiveSOS + ERKPP \rightarrow removedSOS + ERKPP$

12. $activeSOS + ERKPP \rightarrow removedSOS + ERKPP$

13. $cRaf + PKA \rightarrow removedcRaf + PKA$

14. $inactivePKA + PKAA \rightarrow PKA + PKAA$

15. $inactivePKA + Cilostamide \rightarrow PKA + Cilostamide$

16. $PKA \rightarrow inactivePKA$

17. $inactiveEPAC + EPACA \rightarrow EPAC + EPACA$

18. $inactiveEPAC + Cilostamide \rightarrow EPAC + Cilostamide$

19. $EPAC \rightarrow inactiveEPAC$

20. $inactiveRap1 + EPAC \rightarrow activeRap1 + EPAC$

21. $activeRap1 + Gap \rightarrow inactiveRap1 + Gap$

22. $BRaf + activeRap1 \rightarrow BRafPP + activeRap1$

23. $BRafPP \rightarrow BRaf$

24. $MEK + BRafPP \rightarrow MEKPP + BRafPP$

25. $inactiveC3G + boundEGFR \rightarrow activeC3G + boundEGFR$

26. $activeC3G \rightarrow inactiveC3G$

27. $inactiveRap1 + activeC3G \rightarrow activeRap1 + activeC3G$

28. $BRaf + activeRas \rightarrow BRafPP + activeRas$

29. $ERKPP \rightarrow ERK$

30. $boundEGFR \rightarrow degradedEGFR$

B.2.2 Reaction rates

1. $\dfrac{k_1[boundEGFR][inactiveSOS]}{k_1' + [inactiveSOS]}$
2. $k_2[EGF][unboundEGFR] - \; k_{2r}[boundEGFR]$
3. $\dfrac{k_3[activeSOS]}{k_3' + [activeSOS]}$
4. $\dfrac{k_4[activeSOS][inactiveRas]}{k_4' + inactiveRas}$
5. $\dfrac{k_5[Gap][activeRas]}{k_5' + [activeRas]}$
6. $\dfrac{k_6[activeRas][cRaf]}{k_6' + [cRaf]}$
7. $\dfrac{k_7[cRafPP]}{k_7' + [cRafPP]}$
8. $\dfrac{k_8[cRafPP][MEK]}{k_8' + [MEK]}$
9. $\dfrac{k_9[MEKPP]}{k_9' + [MEKPP]}$
10. $\dfrac{k_{10}[MEKPP][ERK]}{k_{10}' + [ERK]}$
11. $\dfrac{k_{11}[ERKPP][inactiveSOS]}{k_{11}' + [inactiveSOS]}$
12. $\dfrac{k_{12}[ERKPP][activeSOS]}{k_{12}' + [activeSOS]}$

13. $\dfrac{k_{13}[PKA][cRaf]}{k'_{13} + [cRaf]}$

14. $\dfrac{k_{14}[PKAA][inactivePKA]}{k'_{14} + [inactivePKA]}$

15. $\dfrac{k_{15}[Cilostamide][inactivePKA]}{k'_{15} + [inactivePKA]}$

16. $\dfrac{k_{16}[PKA]}{k'_{16} + [PKA]}$

17. $\dfrac{k_{17}[EPACA][inactiveEPAC]}{k'_{17} + [inactiveEPAC]}$

18. $\dfrac{k_{18}[Cilostamide][inactiveEPAC]}{k'_{18} + [inactiveEPAC]}$

19. $\dfrac{k_{19}[EPAC]}{k'_{19} + [EPAC]}$

20. $\dfrac{k_{20}[EPAC][inactiveRap1]}{k'_{20} + inactiveRap1}$

21. $\dfrac{k_{21}[Gap][activeRap1]}{k'_{21} + [activeRap1]}$

22. $\dfrac{k_{22}[activeRap1][BRaf]}{k'_{22} + [BRaf]}$

23. $\dfrac{k_{23}[BRafPP]}{k'_{23} + [BRafPP]}$

24. $\dfrac{k_{24}[BRafPP][MEK]}{k'_{24} + [MEK]}$

25. $\dfrac{k_{25}[boundEGFR][inactiveC3G]}{k'_{25} + [inactiveC3G]}$

26. $k_{26}[activeC3G]$

27. $\dfrac{k_{27}[activeC3G][inactiveRap1]}{k'_{27} + [inactiveRap1]}$

28. $\dfrac{k_{28}[activeRas][BRaf]}{k'_{28} + [BRaf]}$

29. $\dfrac{k_{29}[ERKPP]}{k'_{29} + [ERKPP]}$

30. $k_{30}[boundEGFR]$

B.2.3 Species production rates

1. $\dot{[unboundEGFR]} = -k_2[EGF][unboundEGFR] + k_{2r}[boundEGFR]$

2. $\dot{[removedcRaf]} = \frac{k_{13}[PKA][cRaf]}{k'_{13}+[cRaf]}$

3. $\dot{[removedSOS]} = \frac{k_{11}[ERKPP][inactiveSOS]}{k'_{11}+[inactiveSOS]} + \frac{k_{12}[ERKPP][activeSOS]}{k'_{12}+[activeSOS]}$

4. $\dot{[inactiveSOS]} = -\frac{k_1[boundEGFR][inactiveSOS]}{k'_1+[inactiveSOS]} + \frac{k_3[activeSOS]}{k'_3+[activeSOS]} - \frac{k_{11}[ERKPP][inactiveSOS]}{k'_{11}+[inactiveSOS]}$

5. $\dot{[inactiveRas]} = -\frac{k_4[activeSOS][inactiveRas]}{k'_4+[inactiveRas]} + \frac{k_5[Gap][activeRas]}{k'_5+[activeRas]}$

6. $\dot{[inactiveRap1]} = -\frac{k_{20}[EPAC][inactiveRap1]}{k'_{20}+[inactiveRap1]} + \frac{k_{21}[Gap][activeRap1]}{k'_{21}+[activeRap1]} - \frac{k_{27}[activeC3G][inactiveRap1]}{k'_{27}+[inactiveRap1]}$

7. $\dot{[inactivePKA]} = -\frac{k_{14}[PKAA][inactivePKA]}{k'_{14}+[inactivePKA]} - \frac{k_{15}[Cilostamide][inactivePKA]}{k'_{15}+[inactivePKA]} + \frac{k_{16}[PKA]}{k'_{16}+[PKA]}$

8. $\dot{[inactiveEPAC]} = -\frac{k_{17}[EPACA][inactiveEPAC]}{k'_{17}+[inactiveEPAC]} - \frac{k_{18}[Cilostamide][inactiveEPAC]}{k'_{18}+[inactiveEPAC]}$
$+ \frac{k_{19}[EPAC]}{k'_{19}+[EPAC]}$

9. $\dot{[cRafPP]} = \frac{k_6[activeRas][cRaf]}{k'_6+[cRaf]} - \frac{k_7[cRafPP]}{k'_7+[cRafPP]}$

10. $\dot{[cRaf]} = -\frac{k_6[activeRas][cRaf]}{k'_6+[cRaf]} + \frac{k_7[cRafPP]}{k'_7+[cRafPP]} - \frac{k_{13}[PKA][cRaf]}{k'_{13}+[cRaf]}$

11. $\dot{[boundEGFR]} = k_2[EGF][unboundEGFR] - k_{2r}[boundEGFR] - k30[boundEGFR]$

12. $\dot{[activeSOS]} = \frac{k_1[boundEGFR][inactiveSOS]}{k'_1+[inactiveSOS]} - \frac{k_3[activeSOS]}{k'_3+[activeSOS]} - \frac{k_{12}[ERKPP][activeSOS]}{k'_{12}+[activeSOS]}$

13. $\dot{[activeRas]} = \frac{k_4[activeSOS][inactiveRas]}{k'_4+[inactiveRas]} - \frac{k_5[Gap][activeRas]}{k'_5+[activeRas]}$

14. $\dot{[activeRap1]} = \frac{k_{20}[EPAC][inactiveRap1]}{k'_{20}+[inactiveRap1]} - \frac{k_{21}[Gap][activeRap1]}{k'_{21}+[activeRap1]} + \frac{k_{27}[activeC3G][inactiveRap1]}{k'_{27}+[inactiveRap1]}$

15. $\dot{[PKA]} = \frac{k_{14}[PKAA][inactivePKA]}{k'_{14}+[inactivePKA]} + \frac{k_{15}[Cilostamide][inactivePKA]}{k'_{15}+[inactivePKA]} - \frac{k_{16}[PKA]}{k'_{16}+[PKA]}$

16. $\dot{[MEKPP]} = \frac{k_8[cRafPP][MEK]}{k'_8+[MEK]} - \frac{k_9[MEKPP]}{k'_9+[MEKPP]} + \frac{k_{24}[BRafPP][MEK]}{k'_{24}+[MEK]}$

17. $\dot{[MEK]} = -\frac{k_8[cRafPP][MEK]}{k'_8+[MEK]} + \frac{k_9[MEKPP]}{k'_9+[MEKPP]} - \frac{k_{24}[BRafPP][MEK]}{k'_{24}+[MEK]}$

18. $\dot{[ERKPP]} = \frac{k_{10}[MEKPP][ERK]}{k'_{10}+[ERK]} - \frac{k_{29}[ERKPP]}{k'_{29}+[ERKPP]}$

19. $[E\dot{R}K] = -\frac{k_{10}[MEKPP][ERK]}{k_{10}'+[ERK]} + \frac{k_{29}[ERKPP]}{k_{29}'+[ERKPP]}$

20. $[E\dot{P}AC] = \frac{k_{17}[EPACA][inactiveEPAC]}{k_{17}'+[inactiveEPAC]} + \frac{k_{18}[Cilostamide][inactiveEPAC]}{k_{18}'+[inactiveEPAC]} - \frac{k_{19}[EPAC]}{k_{19}'+[EPAC]}$

21. $[E\dot{G}F] = -k_2[EGF][unboundEGFR] + k_{2r}[boundEGFR]$

22. $[BR\dot{a}fPP] = \frac{k_{22}[activeRap1][BRaf]}{k_{22}'+[BRaf]} - \frac{k_{23}[BRafPP]}{k_{23}'+[BRafPP]} + \frac{k_{28}[activeRas][BRaf]}{k_{28}'+[BRaf]}$

23. $[B\dot{R}af] = -\frac{k_{22}[activeRap1][BRaf]}{k_{22}'+[BRaf]} + \frac{k_{23}[BRafPP]}{k_{23}'+[BRafPP]} - \frac{k_{28}[activeRas][BRaf]}{k_{28}'+[BRaf]}$

24. $[acti\dot{v}eC3G] = \frac{k_{25}[boundEGFR][inactiveC3G]}{k_{25}'+[inactiveC3G]} - k_{26}[activeC3G]$

25. $[inacti\dot{v}eC3G] = -\frac{k_{25}[boundEGFR][inactiveC3G]}{k_{25}'+[inactiveC3G]} + k_{26}[activeC3G]$

26. $[degrade\dot{d}EGFR] = k_{30}[boundEGFR]$

27. $[E\dot{P}ACA] = 0$

28. $[G\dot{a}p] = 0$

29. $[P\dot{K}AA] = 0$

30. $[Cilos\dot{t}amide]$

B.2.4 Initial species concentrations

All simulations using the above reactions are performed with the following initial concentrations:

1. $[unboundEGFR]_0 = 500$

2. $[removedcRaf]_0 = 0$

3. $[removedSOS]_0 = 0$

4. $[inactiveSOS]_0 = 1200$

5. $[inactiveRas]_0 = 1200$

6. $[inactiveRap1]_0 = 1200$

7. $[inactivePKA]_0 = 1000$

8. $[inactiveEPAC]_0 = 1000$

9. $[cRafPP]_0 = 0$

10. $[cRaf]_0 = 1500$

11. $[boundEGFR]_0 = 0$

12. $[activeSOS]_0 = 0$

13. $[activeRas]_0 = 0$

14. $[activeRap1]_0 = 0$

15. $[PKA]_0 = 0$

16. $[MEKPP]_0 = 0$

17. $[MEK]_0 = 3000$

18. $[ERKPP]_0 = 0$

19. $[ERK]_0 = 10000$

20. $[EPAC]_0 = 0$

21. $[EGF]_0 = 1000$

22. $[BRafPP]_0 = 0$

23. $[BRaf]_0 = 1500$

24. $[activeC3G]_0 = 0$

25. $[inactiveC3G]_0 = 1200$

26. $[degradedEGFR]_0 = 0$

27. $[EPACA]_0 = 1000$

28. $[Gap]_0 = 2400$

29. $[PKAA]_0 = 1000$

30. $[Cilostamide]_0 = 1000$

Bibliography

[1] F. Al-Awadhi, M. A. Hurn, and C. Jennison. Improving the acceptance rate of the reversible jump MCMC proposals. *Statistics and Probability Letters*, 69:189–198, 2004.

[2] R. Albert. Network inference, analysis, and modeling in systems biology. *The Plant Cell*, 19(11):3327–3338, 2007.

[3] C. Andrieu, N. deFreitas, A. Doucet, and M. I. Jordan. An introduction to MCMC for machine learning. *Machine Learning*, 50:5–43, 2003.

[4] C. Andrieu and E. Moulines. On the ergodicity properties of some adaptive MCMC algorithms. *Annals of Applied Probability*, 16(3):1462–1505, 2006.

[5] C. Andrieu, E. Moulines, and P. Priouret. Stability of stochastic approximation under verifiable conditions. *SIAM Journal Control and Optimization*, 44(1):283–312, 2005.

[6] C. Andrieu and G. O. Roberts. The pseudo-marginal approach for efficient Monte Carlo computations. *The Annals of Statistics*, 37(2):697–725, 2009.

[7] A. Arkin and J. Ross. Statistical construction of chemical reaction mechanisms from measured time-series. *Journal of Physical Chemistry*, (99):970–979, 1995.

[8] S. Arlot and A. Celisse. A survey of cross-validation procedures for model selection. *Statistics Surveys*, 4:40–79, 2010.

[9] R. Asters, B. Borchers, and C. Thurber. *Parameter Estimation and Inverse Problems*. Academic Press, 2004.

[10] M. Bansal, G. D. Gatta, and D. Bernardo. Inference of gene regulatory and compound mode of action from time course gene expression profiles. *Bioinformatics*, 22(7):815–822, 2006.

[11] M. A. Beaumont. Estimation of population growth or decline in genetically monitored populations. *Genetics*, 164:1139–1160, 2003.

[12] J. Berger and L. Pericchi. Objective Bayesian methods for model selection: introduction and comparison. *Model Selection (P.Lahiri, editor), IMS Lecture Notes – Monograph Series*, 38:135–207, 2001.

[13] J. M. Bernardo and A. F. M. Smith. *Bayesian Theory.* Wiley, 2000.

[14] B. Bhattacharjee, D. A. Schwer, P. I. Barton, and Jr. W. H. Green. Optimally-reduced kinetic models: reaction elimination in large-scale kinetic mechanisms. *Combustion and Flame*, 135(3):191–208, 2003.

[15] J. A. Bilmes. A gentle tutorial on the EM algorithm and its application to parameter estimation for Gaussian mixture and hidden Markov models, 1998.

[16] C. M. Bishop. *Pattern Recognition and Machine Learning.* Springer, 2007.

[17] R. Bonneau. Learning biological networks: from modules to dynamics. *Nature Chemical Biology*, 4(11):658–662, 2008.

[18] R. Bonneau, D. J. Reiss, P. Shannon, M. Facciotti, L. Hood, N. S. Baliga, and V. Thorsson. The inferelator: an algorithm for learning parsimonious regulatory networks from systems-biology data sets *de novo*. *Genome Biology*, 7(5):R36, 2006.

[19] K. Braman, T. A. Oliver, and V. Raman. Bayesian analysis of syngas chemistry models. *Combustion Theory and Modelling*, 17(5):858–887, 2013.

[20] S. P. Brooks, P. Giudici, and G. O. Roberts. Efficient construction of reversible jump Markov chain Monte Carlo proposal distributions (with discussion). *Journal of Royal Statistical Society B*, 65:3–39, 2003.

[21] K. P. Burnham and D. R. Anderson. *Model Selection and Multimodal Inference: A Practical Information Theoretic Approach.* Springer, 2nd edition, 2002.

[22] S. C. Burnham, D. P. Searson, M. J. Willis, and A. R. Wright. Inference of chemical reaction networks. *Chemical Engineering Science*, 63(4):862–873, 2008.

[23] O. Cappé, C. P. Robert, and T. Ryden. Reversible jump, birth-and-death and more general continuous time markov chain monte carlo samplers. *Journal of the Royal Statistical Society B*, (65):679–700, 2003.

[24] B. P. Carlin and S. Chib. Bayesian model choice via Markov chain Monte Carlo methods. *Journal of the Royal Statistical Society: Series B (Methodological)*, 57(3):473–484, 1995.

[25] S. Chib. Marginal likelihood from the Gibbs output. *Journal of the American Statistical Association*, 90(432):1313–1321, 1995.

[26] S. Chib and I. Jeliazkov. Marginal likelihood from the Metropolis-Hastings output. *Journal of the American Statistical Association*, 96(453):270–281, 2001.

[27] M. Clyde and E. I. George. Model uncertainty. *Statistical Science*, 19:81–94, 2004.

[28] H. J. Curran, P. Gaffuri, W. J. Pitz, and C. K. Westbrook. A comprehensive modeling study of iso-octane oxidation. *Combustion and Flame*, 129(3):253–280, 2002.

[29] P. J. Davis and P. Rabinowitz. *Methods of Numerical Intergation.* Acedemic Press, 2nd edition, 1984.

[30] P. Dellaportas, J. J. Forster, and I. Ntzoufras. On Bayesian model and variable selection using MCMC. *Statistics and Computing*, 12:27–36, 2002.

[31] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society. Series B (Methodological)*, 39(1):1–38, 1977.

[32] D. G. T. Denison, B. K. Mallick, and A. F. M. Smith. Automatic Bayesian curve fitting. *Journal of the Royal Statistical Society. Series B*, 60(2):333–350, 1998.

[33] O. Deutschmann, R. Schwiedernoch, L. I. Maier, and D. Chatterjee. Comparison between calculated and experimentally determined selectivity and conversion for short-contact-time reactors using honeycomb monoliths. *Natural Gas Conversion VI, Studies in surface science and catalysis, E. Iglesia, J.J. Spivey, T.H. Fleisch (eds.), Elsevier*, 136:215–258, 2001.

[34] I. DiMatteo, C. R. Genovese, and R. E. Kass. Bayesian curve-fitting with free-knot splines. *Biometrika*, 88(4):1055–1071, 2001.

[35] Y. Efendiev, T. Y. Hou, and W. Luo. Preconditioning markov chain monte carlo simulations using coarse scale models. *SIAM Journal on Scientific Computing*, 28:776–803, 2006.

[36] R. S. Ehlers and S. P. Brooks. Adaptive proposal construction for reversible jump MCMC. *Scandinavian Journal of Statistics*, 35:677–690, 2008.

[37] B. Ellis and W. H. Wong. Learning causal Bayesian network structures from experimental data. *Journal of the American Statistical Association*, 103(482):778–789, 2008.

[38] Y. Fan, G. W. Peters, and S. A. Sisson. Automating and evaluating reversible jump MCMC proposal distributions. *Statistics and Computing*, 19:409–421, 2009.

[39] N. Friedman and D. Koller. Being Bayesian about network structure. a Bayesian approach to structure discovery in Bayesian networks. *Machine Learning*, 50:95–125, 2003.

[40] N. Friedman, M. Linial, I. Nachman, and D. Pe'er. Using Bayesian networks to analyze expression data. *Journal of Computational Biology*, 7:601–620, 2000.

[41] N. Friel and A. N. Pettitt. Marginal likelihood estimation via power posteriors. *Journal of the Royal Statistical Society: Series B*, 70(3):589–607, 2008.

[42] N. Galagali and Y. M. Marzouk. Bayesian inference of chemical kinetic models from proposed reactions. *Chemical Engineering Science*, 123:170–190, 2015.

[43] T. S. Gardner, D. Bernardo, D. Lorenz, and J. J. Collins. Inferring genetic networks and identifying compound mode of action via expression profiling. *Science*, 301(5629):102–105, 2003.

[44] T. S. Gardner and J. J. Faith. Reverse engineering transcription control networks. *Physics of Life Reviews*, 2(1):65–88, 2005.

[45] P. H. Garthwaite, J. B. Kadane, and A. O'Hagan. Statistical methods for eliciting probability distributions. *Journal of the American Statistical Association*, 100(470):680–700, 2005.

[46] A. Gelman, J. B. Carlin, H. S. Stern, and D. B. Rudin. *Bayesian Data Analysis.* Chapman and Hall/CRC, 2nd edition, 2004.

[47] A. Gelman and X. L. Meng. Simulating normalizing constants: from importance sampling to bridge sampling to path sampling. *Statistical Science*, 13(2):163–185, 1998.

[48] T. Gerstner and M. Griebel. Numerical integration using sparse grids. *Numerical Algorithms*, (18):209–232, 1998.

[49] J. Geweke. Bayesian inference in econometric models using monte carlo integration. *Econometrica*, (57):1317–1340, 1989.

[50] C. J. Geyer. Markov chain maximum likelihood. In E. Keramigas, editor, *Computing Science and Statistics: The 23rd Symposium on the Interface*. Interface Foundation, Fairfax, 1991.

[51] W. R. Gilks, S. Richardson, and D. J. Spiegelhalter. *Markov chain Monte Carlo in Practice.* Chapman and Hall/ CRC, 1996.

[52] S. J. Godsill. On the relationship between Markov chain Monte Carlo methods for model uncertainty. *Journal of Computational and Graphical Statistics*, 10(2):230–248, 2001.

[53] A. Golightly and D. J. Wilkinson. Bayesian inference for stochastic kinetic models using a diffusion approximation. *Biometrics*, 61(3):781–788, 2005.

[54] D. Goodwin, N. Malalya, H. Moffat, and R. Speth. Cantera: An object-oriented software toolkit for chemical kinetics, thermodynamics, and transport processes, version 2.0.2. available at https://code.google.com/p/cantera/, 2013.

[55] W. P. Gouveia and J. A. Scales. Resolution of seismic waveform inversion: Bayes versus occam. *Inverse Problems*, 13:323–349, 1997.

[56] P. Green and D. Hastie. Reversible jump MCMC. Available at http://www.maths.bris.ac.uk/∼ mapjg/papers/. Access date: 04/24/2014, 2009.

[57] P. J. Green. Reversible jump Markov chain Monte Carlo computation and model determination. *Annals of Stat*, 82(4):711–732, 1995.

[58] P. J. Green. Trans-dimensional Markov chain Monte Carlo. In P. J. Green, N. L. Hjort, and S. Richardson, editors, *Highly Structures Stochastic Systems*. Oxford University Press, Oxford, 2003.

[59] P. J. Green and A. Mira. Delayed rejection in reversible jump Metropolis-Hastings. *Biometrika*, 88:1035–1053, 2001.

[60] U. Grenander and M. I. Miller. Representations of knowledge in complex systems (with discussion). *Journal of the Royal Statistical Society B*, (56):549–603, 1994.

[61] C. Guihenneuc-Jouyaux and J. Rosseau. Laplace expansions in Markov chain Monte Carlo algorithms. *Journal of Computational and Graphical Statistics*, 14(1):75–94, 2005.

[62] J. Hanna, W. Y. Lee, Y. Shi, and A. F. Ghoniem. Fundamentals of electro- and thermochemistry in the anode of solid-oxide fuel cells with hydrocarbon and syngas fuels. *Progress in Energy and Combustion Science*, 40:74–111, 2014.

[63] M. Hartmann, L. Maier, H. D. Minh, and O. Deutschmann. Catalytic partial oxidation of iso-octane over Rhodium catalysts: An experimental, modeling, and simulation study. *Combustion and Flame*, 157(9):1771–1782, 2010.

[64] T. Hastie, R. Tibshirani, and J. Friedman. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer, 2nd edition, 2009.

[65] B. Hayete, T. S. Gardner, and J. J. Collins. Size matters: network inference tackles the genome scale. *Molecular Systems Biology*, 3(77):1–3, 2007.

[66] N. A. Heard, C. C. Holmes, and D. A. Stephens. A quantitative study of gene regulation involved in the immune response of anopheline mosquitoes: an application of bayesian hierarchical clustering of curves. *Journal of the American Statistical Association*, 101:18–26, 2006.

[67] O. Herbinet, W. J. Pitz, and C. K. Westbrook. Detailed chemical kinetic oxidation mechanism for a biodiesel surrogate. *Combustion and Flame*, 154(3):507–528, 2008.

[68] D. Hickman and L. D. Schmidt. Steps in CH_4 oxidation on Pt and Rh surfaces: High-temperature reactor simulations. *AIChE J.*, 39(7):1164–1177, 1993.

[69] A. C. Hindmarsh, P. N. Brown, K. E. Grant, S. L. Lee, R. Serban, D. E. Shumaker, and C. S. Woodward. Sundials: Suite of nonlinear and differential/algebraic equation solvers, 2005.

[70] K. Hukushima and K. Nemoto. Exchange monte carlo method and applicatio to spin glass simulations. *Journal of the Physics Society of Japan*, 65:1604–1608, 1996.

[71] C. Jackson, M. K. Sen, and P. L. Stoffa. An efficient stochastic bayesian approach to optimal parameter and uncertainty estimation for climate model predictions. *Journal of Climate*, 17:2828–2841, 2004.

[72] A. Jasra, D. A. Stephens, and C. C. Holmes. On population-based simulation for static inference. *Statistics and Computing*, 17:263–279, 2007.

[73] C. Ji and S. C. Schmidler. Adaptive Markov chain Monte Carlo for Bayesian variable selection. *J. Comp. and Graph. Stat.*, 22(3):708–728, 2013.

[74] J. Kaipio and E. Somersalo. *Statistical and Computational Inverse Problems*. Springer, 2005.

[75] R. E. Kass and A. E. Raftery. Bayes Factors. *Journal of the American Statistical Association*, 90(430):773–795, 1995.

[76] R. E. Kass, L. Tierney, and J. B. Kadane. The validity of posterior asymptotic expansions based on laplace's method. In S. Geisser, J. S. Hodges, S. J. Press, and A. Zellner, editors, *Bayesian and Likelihood Methods in Statistics and Econometrics*. North-Holland, New York, 1990.

[77] R. J. Kee, M. E. Coltrin, and P. Glarborg. *Chemically Reacting Flow: Theory and Practice*. Wiley, 2003.

[78] M. Komorowski, B. Finkenstadt, C. V. Harper, and D. A. Rand. Bayesian inference of biochemical kinetic parameters using the linear noise approximation. *BMC Bioinformatics*, 10:343–352, 2009.

[79] H. J. Kushner and G. G. Yin. *Stochastic Approximation Algorithms and Applications*. Springer, 2nd edition, 1997.

[80] G. Lawler and A. Sokal. Bounds on the L^2 spectrum for Markov chains and Markov processes. *Transactions of the American Mathematical Society*, 309:557–580, 1988.

[81] F. Liang and W. H. Wong. Real parameter evolutionary monte carlo with applications to bayesian mixture models. *Journal of American Statistical Association*, 96:653–666, 2001.

[82] D. J. C. Mackay. *Information Theory, Inference, and Learning Algorithms*. Cambridge University Press UK, 2003.

[83] D. Madigan, J. York, and D. Allard. Bayesian grpahical models for discrete data. *International Statistical Review*, 63(2):215–232, 1995.

[84] A. A. Margolin, I. Nemenman, K. Basso, C. Wiggins, G. Stolovitzky, R. D. Favera, and A. Califano. Arcane: an algorithm for the reconstruction of gene regulatory networks in a mammalian cellular context. *BMC Bioinformatics*, (7):(Suppl. 1),7, 2006.

[85] N. E. McGuire, N. P. Sullivan, O. Deutschmann, H. Zhu, and R. J. Kee. Dry reforming of methane in a stagnation-flow reactor using Rh supported on strontium-substituted hexaaluminate. *Applied Catalysis A: General*, 394(1–2):257–265, 2011.

[86] N. E. McGuire, N. P. Sullivan, R. J. Kee, H. Zhu, A. J. Nabity, J. R. Engel, D. T. Wickham, and M. J. Kaufman. Catalytic steam reforming of methane using Rh supported on Sr-substituted hexaaluminate. *Chemical Engineering Science*, 64(24):5231–5239, 2009.

[87] X. L. Meng and W. H. Wong. Simulating ratios of normalizing constants via a simple identity: a theoretical exploration. *Statistical Sinica*, (6):831–860, 1996.

[88] T. J. Mitchell and J. J. Beauchamp. Bayesian variable selection in linear regression. *Journal of the American Statistical Association*, 83(404):1023–1032, 1988.

[89] A. Mohammad-Djafari. Bayesian inference for inverse problems. In *Bayesian Inference and Maximum Entropy Methods in Science and Engineering*, volume 21, pages 477–496. 2002.

[90] S. Mukherjee and T. P. Speed. Network inference using informative priors. *Proceedings of the National Academy of Sciences*, 105(38):14313–14318, 2008.

[91] J. D. Murray. *Mathematical Biology: I. An Introduction.* Springer, 3rd edition, 2002.

[92] I. Nachman, A. Regev, and N. Friedman. Inferring quantitative models of regulatory networks from expression data. *Bioinformatics*, pages i248–i256, 2004.

[93] R. M. Neal. Annealed importance sampling. *Statistics and Computing*, (11):125–139, 2001.

[94] M. A. Newton and A. E. Raftery. Approximate Bayesian inference with the weighted likelihood bootstrap. *Journal of Royal Statistical Society: Series B*, 56(1):3–48, 1994.

[95] C. J. Oates, B. T. Hennessy, Y. Lu, G. B. Mills, and S. Mukherjee. Network inference using steady-state data and Goldbeter-Koshland kinetics. *Bioinformatics*, 28(18):2342–2348, 2012.

[96] C. J. Oates and S. Mukherjee. Network inference and biological dynamics. *Annals of Applied Statistics*, 6(3):1209–1235, 2012.

[97] O. O. Oluwole, B. Bhattacharjee, J. E. Tolsma, P. I. Barton, and W. H. Green Jr. Rigorous valid ranges for optimally-reduced kinetic models. *Combustion and Flame*, 146(1–2):348–365, 2006.

[98] J. Omony. Biological network inference: A review of methods and assessment of tools and techniques. *Annual Research and Review in Biology*, 4(4):577–601, 2014.

[99] J. Prager, H. Najm, and J. Zador. Uncertainty quantification in the ab initio rate-coefficient calculation for the $CH_3CH(OH)CH_3$ + OH → $CH_3C(OH)CH_3$ + H_2O reaction. *Proceedings of the Combustion Institute*, 34(1):583–590, 2013.

[100] C. J. Preston. Spatial birth-and-death processes. *Bulletin of the International Statistical Institute*, (46):371–391, 1977.

[101] J. K. Pritchard, M. Stephens, and P. Donnelly. Inference of population structure using multilocus genotype data. *Genetics*, 155:945–959, 2001.

[102] A. E. Raftery, D. Madigan, and J. A. Hoeting. Bayesain model averaging for linear regression models. *Journal of the American Statistical Association*, 92(437):179–191, 1997.

[103] B. D. Ripley. Modelling spatial patterns (with discussion). *Journal of the Royal Statistical Society B*, (39):172–212, 1977.

[104] H. Robbins and S. Monro. A stochastic approximation method. *Annals of Mathematical Statistics*, 22(3):400–407, 1951.

[105] C. P. Robert and G. Casella. *Monte Carlo Statistical Methods*. Springer, 2nd edition, 2004.

[106] G. O. Roberts and J. S. Rosenthal. Coupling and ergodicity of adaptive Markov chain Monte Carlo algorithms. *Journal of Applied Probability*, 44(1):458–475, 2007.

[107] K. Sachs, D. Gifford, T. Jaakkola, P. Sorger, and D. A. Lauffenburger. Bayesian network approach to cell signaling pathway modeling. *Science STKE*, 148:pe38, 2002.

[108] B. Schoeberl, C. Eichler-Jonsson, E. D. Gilles, and G. Müller. Computational modeling of the dynamics of the map kinase cascade activated by surface and internalized egf receptors. *Nature Biotechnology*, 20:370–375, 2002.

[109] G. E. Schwarz. Estimating the dimension of a model. *Annals of Statistics*, 6(2):461–464, 1978.

[110] S. A. Sisson. Transdimensional markov chains: A decade of progress and future perspectives. *Journal of the American Statistical Association*, 100(471):1077–1089, 2005.

[111] D. S. Sivia. *Data Analysis: A Bayesian Tutorial.* Oxford University Press USA, 2nd edition, 2006.

[112] M. Stephens. Bayesian analysis of mixture models with an unknown number of components - an alternative to reversible jump methods. *The Annals of Statistics*, (28):40–74, 2000.

[113] A. Tarantola. *Inverse Problem Theory and Methods for Model Parameter Estimation.* SIAM, 2005.

[114] L. Tierney. Markov chains for exploring posterior distributions. *Annals of Statistics*, 22(4):1701–1728, 1994.

[115] A. Warnes. The normal kernel coupler: an adaptive markov chain monte carlo method for efficiently sampling from multimodal distributions, 2007.

[116] A. V. Werhli and D. Husmeier. Gene regulatory network reconstruction by Bayesian integration of prior knowledge and/or different experimental conditions. *Journal of Bioinformatics and Computational Biology*, 6(3):543–572, 2008.

[117] C. F. J. Wu. On the convergence properties of the EM algorithm. *The Annals of Statistics*, 11(1):95–103, 1983.

[118] T.-R. Xu, V. Vyshemirsky, A. Gormand, A. von Krigsheim, M. Girolami, G. S. Baillie, D. Ketley, A. J. Dunlop, G. Milligan, M. D. Houslay, and W. Kolch. Inferring signaling pathway topologies from multiple perturbation measurements of specific biochemical species. *Science Signaling*, 3(113):ra20, 2010.