New Advances on Bayesian and Decision-Theoretic Approaches for Interactive Machine Learning

Hoang Trong Nghia

(B.Sc. (Hons.), VNU)

A thesis submitted for the degree of

Doctor of Philosophy

Department of Computer Science, School of Computing

National University of Singapore

2014

New Advances on Bayesian and Decision-Theoretic Approaches
for Interactive Machine Learning

by

Hoang Trong Nghia

Declaration

I hereby declare that the thesis is my original work and it has been written by me in its entirety. I have duly acknowledged all the sources of information which I have been used in the thesis. This thesis has also not been submitted for any degree in any university previously.

Hoang Trong Nghia

20 October 2014

Publications

1. Publications during candidature:

- **Decision-Theoretic Approach to Maximizing Observation of Multiple Targets in Multi-Camera Surveillance**.
  Prabhu Natarajan, Trong Nghia Hoang, Kian Hsiang Low and Mohan Kankanhalli. In *Proceedings of the* 11^{th} *International Conference on Autonomous Agents and Multi-Agent Systems (AAMAS-12)*, pages $155 - 162$, Valencia, Spain, June $4 - 8$, 2012.

- **Decision-Theoretic Coordination and Control for Active Multi-Camera Surveillance in Uncertain, Partially Observable Environments**.
  Prabhu Natarajan, Trong Nghia Hoang, Kian Hsiang Low and Mohan Kankanhalli. In *Proceedings of the* 6^{th} *ACM/IEEE International Conference on Distributed Smart Cameras (ICDSC-12)*, pages $1 - 6$, Hong Kong, October 30 - November 2, 2012.

- **Interactive POMDP Lite: Towards Practical Planning to Predict and Exploit Intentions for Interacting with Self-Interested Agents**.
  Trong Nghia Hoang and Kian Hsiang Low. In *Proceedings of the* 23^{rd} *International Joint Conference on Artificial Intelligence (IJCAI-13)*, pages $2298 - 2305$, Beijing, China, August $3 - 9$, 2013.

- **A General Framework for Interacting Bayes-Optimally with Self-Interested Agents using Arbitrary Parametric Model and Model Prior**.
  Trong Nghia Hoang and Kian Hsiang Low. In *Proceedings of the* 23^{rd} *International Joint Conference on Artificial Intelligence (IJCAI-13)*, pages $1394 - 1400$, Beijing, China, August $3 - 9$, 2013.

- **Nonmyopic ϵ-Bayes-Optimal Active Learning of Gaussian Processes**.
  Trong Nghia Hoang, Kian Hsiang Low, Patrick Jaiilet and Mohan Kankanhalli. In *Proceedings of the* 31^{st} *International Conference on Machine Learning (ICML-14)*, pages $739 - 747$, Beijing, China, June $21 - 26$, 2014. Also appeared in *RSS-14 Workshop on Non-Parametric Learning in Robotics*, Berkeley, CA, July 12, 2014.

- **Active Learning is Planning: Nonmyopic ϵ-Bayes-Optimal Active Learning of Gaussian Processes**.

  Trong Nghia Hoang, Kian Hsiang Low, Patrick Jaillet and Mohan Kankanhalli. In T. Calders, F. Esposito, E. Hüllermeier, R. Meo, editors, *Proceedings of the 7^{th} European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases (ECML/PKDD-14) Nectar (New Scientific and Technical Advances in Research) Track*, Part III, LNCS 8726, pages $499 - 503$, Springer, Heidelberg, Nancy, France, September $15 - 19$, 2014.

- **Scalable Decision-Theoretic Coordination and Control for Real-time Active Multi-Camera Surveillance**.

  Prabhu Natarajan, Trong Nghia Hoang, Yongkang Wong, Kian Hsiang Low and Mohan Kankanhalli. Accepted for publication in *Proceedings of the 8^{th} ACM/IEEE International Conference on Distributed Smart Cameras (ICDSC-14)* (Invited Paper to Special Session on Smart Cameras for Smart Environments), Venezia, Italy, November $4 - 7$, 2014.

- **Recent Advances in Scaling up Gaussian Process Predictive Models for Large Spatiotemporal Data**.

  Kian Hsiang Low, Jie Chen, Trong Nghia Hoang, Xu Nuo and Patrick Jaillet. Accepted for publication in *Proceedings of the Dynamic Data-driven Environmental Systems Science Conference (DyDESS-14)*, MIT, Cambridge, MA, November $5 - 7$, 2014.

2. Publications included in this thesis:

- **A General Framework for Interacting Bayes-Optimally with Self-Interested Agents using Arbitrary Parametric Model and Model Prior**.

  Trong Nghia Hoang and Kian Hsiang Low. In *Proceedings of the 23^{rd} International Joint Conference on Artificial Intelligence (IJCAI-13)*, pages $1394 - 1400$, Beijing, China, August $3 - 9$, 2013 (Chapter 3).

- **Nonmyopic ϵ-Bayes-Optimal Active Learning of Gaussian Processes**.
  Trong Nghia Hoang, Kian Hsiang Low, Patrick Jaiilet and Mohan Kankanhalli. In *Proceedings of the* 31^{st} *International Conference on Machine Learning (ICML-14)*, pages $739 - 747$, Beijing, China, June $21 - 26$, 2014. Also appeared in *RSS-14 Workshop on Non-Parametric Learning in Robotics*, Berkeley, CA, July 12, 2014 (Chapter 4).

- **A Unifying Framework of Anytime Sparse Gaussian Process Regression Models**.
  Trong Nghia Hoang, Quang Minh Hoang and Kian Hsiang Low.
  (Chapter 5, to be submitted for review).

Abstract

The exploration-exploitation trade-off is a fundamental dilemma in many interactive learning scenarios which include both aspects of reinforcement learning (RL) and active learning (AL): An autonomous agent, situated in an unknown environment, has to actively extract knowledge from the environment by taking actions (or conducting experiments) based on its previously collected information to make accurate predictions or to optimize some utility functions. Thus, to make the most effective use of their resource-constrained budget (e.g., processing time, experimentation cost), the agent must choose carefully between (a) exploiting options (e.g., actions, experiments) which are recommended by its current, possibly incomplete model of the environment, and (b) exploring the other *ostensibly* sub-optimal choices to gather more information.

For example, an RL agent has to face a dilemma between (a) exploiting the most-rewarding action according to the current statistical model of the environment at the risk of running into catastrophic situations if the model is not accurate, and (b) exploring a sub-optimal action to gather more information so as to improve the model's accuracy at the potential price of losing the short-term reward. Similarly, an AL algorithm/agent has to consider between (a) conducting the most informative experiments according to its current estimation of the environment model's parameters (i.e., exploitation), and (b) running experiments that help improving the estimation accuracy of these parameters (i.e., exploration).

More often, learning strategies that ignore exploration will likely exhibit sub-optimal performance due to their imperfect knowledge while, conversely, those that entirely focus on exploration might suffer the cost of learning without benefitting from it. Therefore, a good exploration-exploitation trade-off is critical to the success of those interactive learning agents: In order to perform well, they must strike the right balance between these two conflicting objectives. Unfortunately, while this trade-off has been well-recognized since the early days of RL, the studies of exploration-exploitation have mostly been developed for theoretical settings in the respective field of RL and, perhaps surprisingly, glossed over in the existing AL literature. From a practical point of view, we see three limiting factors:

1. Previous works addressing the exploration-exploitation trade-off in RL have largely focused on simple choices of the environment model and consequently, are not practical enough to accommo-

date real-world applications that have far more complicated environment structures. In fact, we find that most recent advances in Bayesian reinforcement learning (BRL) have only been able to analytically trade off between exploration and exploitation under a simple choice of models such as Flat-Dirichlet-Multinomial (FDM) whose independence and modeling assumptions do not hold for many real-world applications.

2. Nearly all of the notable works in the AL literature primarily advocate the use of *greedy/myopic* algorithms whose rates of convergence (i.e., the number of experiments required by the learning algorithm to achieve a desired performance in the worst case) are provably *minimax optimal* for simple classes of learning tasks (e.g., threshold learning). While these results have greatly advanced our understanding about the limit of *myopic* AL in worst-case scenarios, significantly less is presently known about whether it is possible to devise *nonmyopic* AL strategies which optimize the exploration-exploitation trade-off to achieve the best expected performance in budgeted learning scenarios.

3. The issue of scalability of the existing predictive models (e.g., Gaussian processes) used in AL has generally been underrated since the majority of literature considers *small-scale* environments which only consist of a few thousand candidate experiments to be selected by *single-mode* AL algorithms one at a time prior to retraining the model. In contrast, *large-scale* environments usually have a massive set of million candidate experiments among which tens or hundreds of thousands should be actively selected for learning. For such data-intensive problems, it is often more cost-effective to consider *batch-mode* AL algorithms which select and conduct multiple experiments in parallel at each stage to collect observations in batch. Retraining the predictive model after incorporating each batch of observations then becomes a computational bottleneck as the collected dataset at each stage quickly grows up to hundreds of thousand or even millions of data points.

This thesis outlines some recent progresses that we have been able to make while working toward satisfactory answers to the above challenges, along with practical algorithms that achieve them:

1. In particular, in order to put BRL into practice for more complicated and practical problems, we propose a novel framework called *Interactive Bayesian Reinforcement Learning* (I-BRL) to integrate the general class of parametric models and model priors, thus allowing the practitioners' domain

knowledge to be exploited to produce a fine-grained and compact representation of the environment as often required in many real-world applications. Interestingly, we show how the nonmyopic Bayes-optimal policy can be derived analytically by solving I-BRL exactly and propose an approximation algorithm to compute it efficiently in polynomial time. Our empirical studies show that the proposed approach performs competitively with the existing state-of-the-art algorithms.

2. Then, to establish a theoretical foundation for the exploration-exploitation trade-off in single-mode active learning scenarios with resource-constrained budgets, we present a novel $\underline{\epsilon}$-*$\underline{B}$ayes-optimal Decision-Theoretic $\underline{A}$ctive $\underline{L}$earning* (ϵ-BAL) framework which advocates the use of differential entropy as a performance measure and consequently, derives a learning policy that can approximate the optimal expected performance arbitrarily closely (i.e., within an arbitrary loss bound ϵ). To meet the real-time requirement in time-critical applications, we then propose an asymptotically ϵ-optimal, branch-and-bound anytime algorithm based on ϵ-BAL with performance guarantees. In practice, we empirically demonstrate with both synthetic and real-world datasets that the proposed approach outperforms the state-of-the-art algorithms in budgeted scenarios.

3. Lastly, to facilitate the future developments of *large-scale, nonmyopic* AL applications, we further introduce a highly scalable family of *anytime* predictive models for AL which provably converge toward a well-known class of sparse Gaussian processes (SGPs). Unlike the existing predictive models of AL which cannot be updated incrementally and are only capable of processing middle-sized datasets (i.e., a few thousands of data points), our proposed models can process massive datasets in an *anytime* fashion, thus providing a principled trade-off between the processing time and the predictive accuracy. The efficiency of our framework is then demonstrated empirically on a variety of large-scale real-world datasets; one of which contains more than 2 millions data points.

Acknowledgements

I would like to express my heartfelt thanks to my advisor, A/Prof. Low Kian Hsiang, who inspired (and contributed to) this research, for supporting and encouraging me with his timely advices and insightful discussions.

My greatest gratitude is also extended to Prof. Mohan Kankanhalli, my thesis committee members, Prof. David Hsu, Prof. Tan Chew Lim, A/Prof. Fabio Ramos, A/Prof. Lee Wee Sun and A/Prof. Leong Tze Yun, for devoting time and effort to read this thesis and its preliminary version as well as providing constructive comments. I would like to specifically thank Prof. Mohan Kankanhalli for funding my research and allowing me to pursue my interested topics for my PhD degree.

On a personal note, I am eternally grateful to my parents, for helping me establish and pursue my academic interests while growing up, and my special one, Thanh Tuyen, for giving me a life full of love and laughs during my stay in Singapore. This thesis would never be completed without their constant and unwavering support.

Lastly, I would like to thank my brother for helping me deliver the final work that completes this thesis. I hope this exposure to research would further inspire you to pursue an academic career in computer science and I look forward to collaborating with you again in the future.

This thesis is dedicated to my family.
Their unconditional love and support is invaluable.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Motivation

Interactive learning has recently emerged as an increasingly important focal theme in machine learning which investigates how autonomous agents (e.g., robots, software programs) may come to operate intelligently by interacting (or experimenting) with their unknown physical (virtual) environments (Fig. 1.1) and possibly, other self-interested entities (e.g., humans). This includes both aspects of active learning (AL) and reinforcement learning (RL) in which an intelligent agent strives to learn the hidden structure of the environment to conceive effective operating policies given a resource-constrained budget of interaction (e.g., experimentation cost, mission time).

Therefore, to learn efficiently within the allowed budget, the agent must be proactive in planning its actions (or conducting its experiments) to extract the most informative feedbacks from the environment. Specifically, these feedbacks are usually provided in terms of empirical observations, corrective evaluations (e.g., active learning) or numerical rewards (e.g., reinforcement learning) that encourage or discourage the

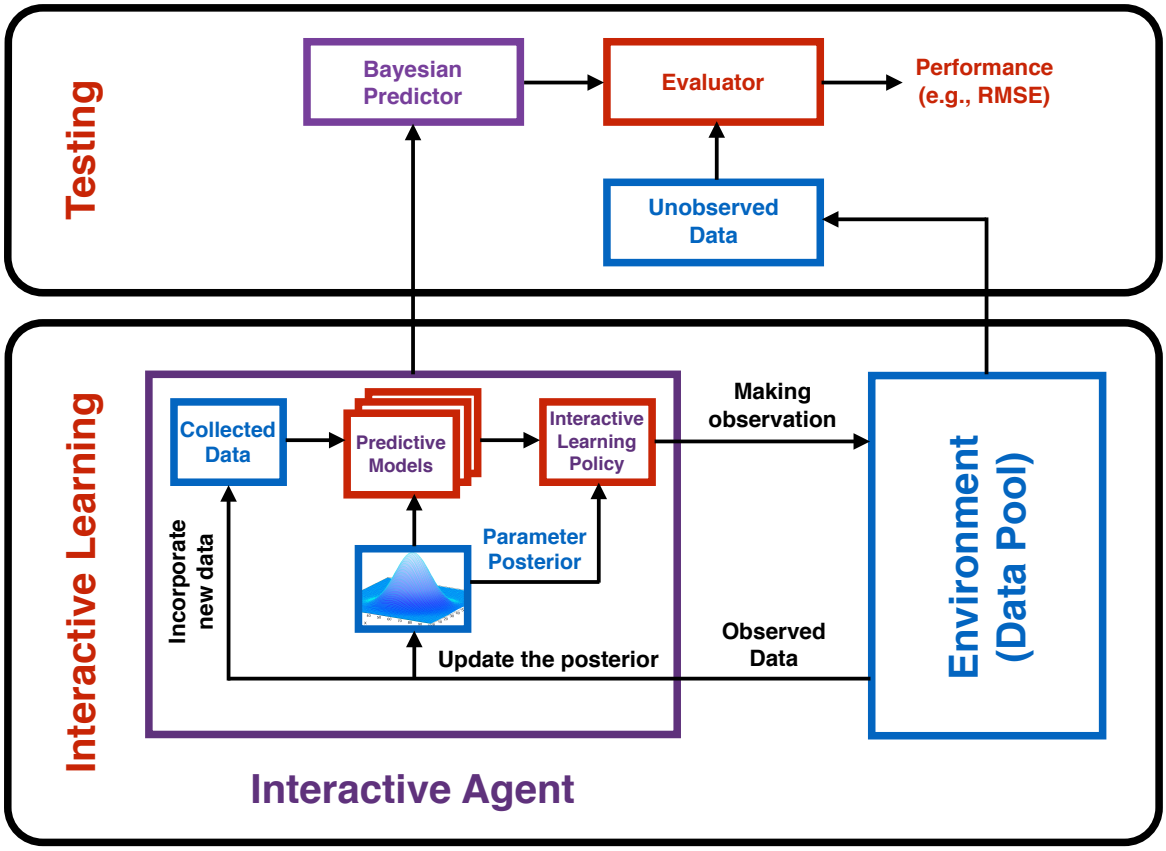


Figure 1.1: Interactive learning system.

agents' current behaviors and hence, provide more information to help them adjust and decide on the next course of actions. Consequently, this motivates an optimization approach to the original learning problem: In order to achieve its goal effectively subject to practical resource constraints, the learning agent needs to compute an interactive policy that maximizes its expected performance in terms of the total rewards given by the environment or some internal performance metrics which are used to measure its learning progress (e.g., entropy-based active sampling).

Interestingly, to maximize its expected performance, an interactive learning agent has to face an exploration-exploitation dilemma: Exploiting options recommended by its current and possibly incomplete knowledge vs. exploring sub-optimal options that could possibly lead to the discovery of new knowledge. In particular, exploitative approaches that always take actions which are most rewarding (or conduct experiments which are most informative) according to the current statistical model of the environment are often inadequate and sub-optimal as they are exposed to the risk of running into catastrophic situations (or wasting resource on uninformative experiments) if the model is inaccurate. On the other hand, approaches which are too vigorous in exploring the other sub-optimal choices of actions or experiments will most likely suffer the cost of learning without benefitting from it: If the agent uses too much resource on redundant exploration, the remaining budget might not be sufficient to exploit what it has successfully learnt. Thus, this trade-off is therefore critical to the success of such interactive learning agent: In order to perform well, it must strike the right balance between these two resource-competing objectives.

To elaborate, let us consider, for example, reinforcement learning (RL) scenarios in which an agent strives to learn the latent dynamics of the environment to maximize its accumulated reward [Poupart *et al.*, 2006] by sequentially planning which action to take at each step. The RL agent then has to choose between (a) taking the best action according to the current statistical model of the environment at the risk of losing an opportunity to recognize a better action if the model is not accurate (i.e., exploitation) vs. (b) choosing other sub-optimal actions to explore and consequently improving the model's accuracy at the potential price of losing the short-term rewards (i.e., exploration). Similarly, in environmental sensing, an active learning (AL) algorithm has to sequentially decide the most informative locations to be sampled for minimizing the predictive uncertainty of the unobserved areas of a phenomenon given

a sampling budget [Low *et al.*, 2008; Low *et al.*, 2009]. In particular, if the model parameters of the phenomenon are not known a priori, the learning algorithm's predictive performance then depends on how informative the collected observations are for both parameter estimation as well as prediction given the true parameters. However, it has also been revealed in previous studies that sampling data that is efficient for parameter estimation is not necessarily efficient for prediction [Müller, 2007; Martin, 2001]. This consequently leads to a potential trade-off between exploration and exploitation which bears a striking resemblance to the above exploration-exploitation dilemma in RL: (a) sampling at the most informative location according to the current, possibly inaccurate estimation of the model's parameters (i.e., exploitation) vs. (b) choosing locations that provide more information about the latent parameters (i.e., exploration).

Examples of such interactive learning systems include:

- **Reinforcement Learning (RL).** An autonomous vehicle that learns to adjust its acceleration and steering behavior properly depending on its observations of the other human-driven vehicles [Wang *et al.*, 2012; Hoang and Low, 2013a], a team of robots playing soccer that learns to coordinate their ball kicks with respect to the locations of their opponents [Stone *et al.*, 2005; Riedmiller *et al.*, 2009], spoken dialog management systems where software agents interactively participate in conversation with human users [Williams, 2006; Li *et al.*, 2009], and online recommendation systems (e.g., ads, news, music) [Li *et al.*, 2010] that learn to decide which content to show to maximize the revenue/reward over time based on the previously collected user statistics (e.g., users' clicks).

- **Active Learning (AL).** Environmental sensing and monitoring applications (e.g., precision agriculture [Tokekar *et al.*, 2013], mineral prospecting [Low *et al.*, 2007], monitoring of ocean and freshwater phenomena like harmful algal bloom [Leonard *et al.*, 2007; Dolan *et al.*, 2009; Podnar *et al.*, 2010], forest ecosystems, pollution or contamination) where a high-resolution *in situ* sampling of the spatial phenomenon of interest is impractical due to prohibitively costly sampling budget requirements (e.g., number of deployed sensors, energy consumption): For such applications, it is therefore desirable to select and conduct experiments (e.g., deploying sensors to make observations) at the most informative locations within the area of interest to model and predict the phenomenon accurately.

Unfortunately, although this trade-off has been recognized since the early days of RL, the studies of exploration-exploitation have been developed mostly for theoretical settings in the respective field of RL and almost glossed over in the literature of AL. To the extent of our knowledge, the only work in AL which explicitly attempt to address the exploration-exploitation trade-off is that of Krause and Guestrin [2007]. In particular, we can identify the following limiting factors that need to be addressed to facilitate future developments of interactive learning in general:

1. Previous works addressing the exploration-exploitation trade-off in RL has tacitly assumed very simple and specific parameterizations of the unknown environments, thus rendering themselves inapplicable to many practical problems where the corresponding environments often have a far more complicated parametric structure [Hoang and Low, 2013a]. In fact, while there exists a broad range of principled frameworks addressing this trade-off in Bayesian reinforcement learning (BRL), most

of these works have been developed for simple choices of environment models such as Flat-Dirichlet-Multinomial (FDM) [Poupart *et al.*, 2006] which is inadequate to model the environment in many real-world applications [Chen *et al.*, 2012; Low *et al.*, 2011; Low *et al.*, 2012; Cao *et al.*, 2013]. For example, in self-interested multi-agent environments, the transition dynamics are mainly controlled by the other agent's stochastic behavior for which FDM's independence and modeling assumptions do not hold [Hoang and Low, 2013b; Natarajan *et al.*, 2012a; Natarajan *et al.*, 2012b] and consequently, prevent its behavior from being generalized across different states [Asmuth and Littman, 2011; A.-Lopez *et al.*, 2012] nor specified precisely using prior domain knowledge. In addition, the other agent's behavior often needs to be modeled differently depending on the specific context [Hoang and Low, 2013a]. Thus, grounding in the context of existing BRL frameworks, either the domain expert struggles to best fit his prior knowledge to the supported set of parameterizations or the agent developer has to re-design the framework to incorporate a new modeling scheme. Arguably, there is no free lunch when it comes to modeling the agent's behavior across various applications.

2. Most of the notable works [Hanneke, 2007; Balcan *et al.*, 2009; Yang *et al.*, 2011; Dasgupta *et al.*, 2007; Beygelzimer *et al.*, 2009] in the AL literature advocate the use of greedy/myopic algorithms whose rates of convergence (i.e., the number of experiments required for the learner to achieve a desired performance in the worst case) are provably minimax optimal with respect to some simple classes of learning tasks (i.e., threshold learning) or nonmyopic approaches that tackle exploration and exploitation separately [Krause and Guestrin, 2007] and consequently, exhibit sub-optimal behavior in the presence of budget constraints. Although these works often come with competitive *worst-case* performance guarantees which are theoretically interesting, significantly less is presently known about how well the existing algorithms perform

on average: Specifically, how good are their *expected* performance in practical, complex environmental domains, and more importantly, whether it is possible to to devise *nonmyopic* AL strategies which *jointly optimize* the exploration-exploitation trade-off to achieve the best expected performance in such budgeted learning scenarios? Naively, one might be tempted to frame active learning as a sequential decision problem that jointly and naturally optimizes the above exploration-exploitation trade-off while maintaining a Bayesian belief over the model parameters. However, such a nonmyopic Bayes-optimal policy, unlike its counterpart in FDM-based BRL [Poupart *et al.*, 2006], cannot be analytically derived in AL contexts for which the model parameters are unknown and the classical RL's discrete-state, Markov assumptions do not hold [Solomon and Zacks, 1970].

3. While there exists a clear conscience among researchers on the issue of scalability of AL algorithms when applying to practical domains, the scalability of their underlying predictive models has unfortunately been underrated since the majority of literature has only considered *small-scale* domains for which each conducted experiment returns a single observation sample (i.e., single-mode AL). This only amounts to a dataset of moderate size, assuming the active budget is restricted to a few hundreds of experiments, which can be handled comfortably using the existing ML models. The main effort of the existing AL literature is, therefore, mainly devoted to devising computationally efficient learning policies instead of scaling up their underlying learning models to process larger datasets.

However, there also exists many *large-scale* practical settings for which retraining these predictive model (e.g., Gaussian processes) after incorporating new observations is very computational expensive, especially if they do not support efficient incremental update. Imagine, for example, an autonomous underwater vehicle which sample a

batch of observations each time it takes a dive: Its collected dataset thus quickly grows up to a very large size since each diving action may collect thousands of observations instead of a single one. Alternatively, in high-throughput experimental designs such as crowdsourcing annotation [Sabou *et al.*, 2014], product marketing [Kemple *et al.*, 2003], resource allocation [Golovin and Krause, 2011] and vaccination in epidemiology [Anshelevich *et al.*, 2009], it is often more cost-effective to choose multiple actions to be carried out in parallel and receive feedbacks in batch which inform the next set of actions [Chen and Krause, 2013]. This is typically known as batch-mode AL which also increases the size of the collected dataset tremendously and as a result, render the traditional AL predictive models computationally impractical.

1.2 Objective

The main focus of this thesis is thus to address the following question:

> Given a resource-constrained budget for interaction, how then does an interactive learning agent optimize the trade off between exploration and exploitation in practical, complex environmental domains efficiently?

The following critical issues arise in answering this question, all of which reflect the serious limitations of existing RL and AL algorithms in many practical domains of applications (Section 1.1):

- **Practical Learning Model for RL.** How can existing BRL frameworks be refined to allow a domain expert to freely incorporate his choice of design in modeling the other agents' behaviors? This question is significant in putting theory into practice and, when answered, can additionally bridge the gap between learning in single- and (self-interested) multi-agent systems.

- **Exploration vs. Exploitation in AL.** How can the notion of Bayes-optimality in RL be exploited for AL problems where the model parameters are unknown and the classical RL's discrete-state, Markov assumptions do not hold? More importantly, is it possible to formulate and tractably derive the Bayes-optimal policy in such *non-Markovian* environments to circumvent the exploitation-exploration dilemma in principle?

- **Scalable Learning Model for AL.** How do we design *scalable* learning models for existing AL algorithms to facilitate its future developments in *large-scale, data-intensive* applications? Specifically, is it possible to design models that efficiently support incremental update in an anytime fashion, thus providing a principled trade-off between the processing time and the learning accuracy?

The above-mentioned issues are then considered and resolved in the development of this thesis as described next.

1.3 Contributions

The work in this thesis supports the following statements:

> Existing BRL frameworks can be generalized to robustly accommodate any parametric model and model prior, thus bridging the gap in putting BRL into practice for more realistic and practical tasks such as learning in self-interested multi-agent systems.
>
> Using differential entropy as an internal performance measure for an AL agent, it is then possible to maintain a balance between exploration and exploitation that guarantees an ϵ-Bayes-optimal expected performance with respect to an arbitrary, user-defined loss bound ϵ.

> It is possible to construct a family of anytime learning frameworks for the existing AL algorithms which efficiently approximates and provably converges towards a well-known class of Bayesian non-parametric models (e.g., Gaussian processes). The constructed models are capable of processing massive datasets containing millions of data points in an anytime fashion which naturally trades off between processing time and learning accuracy.

All of these claims are substantiated by the following novel contributions which are summarized below:

1. Interactive Bayesian Reinforcement Learning (Chapter 3):

   - We present a novel generalization of BRL called Interactive BRL (I-BRL) to integrate any parametric model and model prior of the environment specified by domain experts (Section 3.1), consequently yielding two advantages: The environment can be represented (a) in a fine-grained manner based on the practitioners' prior domain knowledge, and (b) compactly to be generalized across different states, thus overcoming the limitations of FDM.

   - In particular, we show how the nonmyopic Bayes-optimal policy can be derived analytically by solving I-BRL exactly (Section 3.2.1) and propose an approximation algorithm to compute it efficiently in polynomial time (Section 3.2.2).

   - Empirically, we demonstrate I-BRL's performance via a set of benchmark problems as well as an interesting traffic problem modeled after a real-world situation (Section 3.3).

   - For interested readers, we discuss the existing BRL literature in Section 2.1. Their strengths and weaknesses are highlighted in comparison to I-BRL.

2. Nonmyopic ϵ-Bayes-Optimal Active Learning (Chapter 4):

- We present an efficient decision-theoretic planning approach to nonmyopic active sensing/learning that can still preserve and exploit the principled Bayesian sequential decision problem framework for jointly and naturally optimizing the exploration-exploitation trade-off (Section 4.2.1) and consequently does not incur the limitations of the algorithm of Krause and Guestrin [2007].
- Although the exact Bayes-optimal policy to the active sensing problem cannot be derived [Solomon and Zacks, 1970], we show that it is in fact possible to solve for a nonmyopic ϵ-*Bayes-optimal active learning* (ϵ-BAL) policy (Sections 4.2.2 and 4.2.3) given a user-defined bound ϵ. In other words, our proposed ϵ-BAL policy can approximate the optimal expected active sensing performance arbitrarily closely (i.e., within an arbitrary loss bound ϵ). In contrast, the algorithm of Krause and Guestrin [2007] can only yield a sub-optimal performance bound[1].
- To meet the real-time requirement in time-critical applications, we then propose an asymptotically ϵ-optimal, branch-and-bound anytime algorithm based on ϵ-BAL with performance guarantee (Section 4.2.4).
- We empirically demonstrate using both synthetic and real-world datasets that, with limited budget, our proposed approach outperforms state-of-the-art algorithms (Section 4.3).
- For the readers' reference, we discuss and review the existing AL literature in Section 2.2 to highlight their strengths and weaknesses in comparison to our proposed framework's.

[1]Its induced policy is guaranteed not to achieve worse than the optimal performance by more than a factor of $1/e$.

3. Scalable Predictive Modeling Platforms for Active Learning (Chapter 5):

   - We introduces a novel framework of *inverse variational inference* to theoretically derive a non-trivial, concave objective functional (of distributions) whose optimum coincides with the predictive distribution of a particular user-specified SGP model (Section 5.2). This effectively allow us to construct an alternative anytime numerical computation of the selected SGP model by iteratively following the stochastic gradient of the objective function.

   - Specifically, we show that if the selected SGP model exhibits certain conditional independence structures, the derived stochastic gradient does not depend on the number of data points, thus making the time complexity of each update iteration independent of the size of data (Section 5.2.2).

   - We further identify and prove that such necessary conditional independence structures are in fact satisfied by a very well-known class of low-rank covariance approximation SGP models (Section 5.2.1). This results in an anytime learning framework capable of processing millions of data points in a single-core machine. For comparison, interested readers are referred to Section 2.3 for a detailed discussion on the computational efficiency of the existing state-of-the-art SGP approaches on big data.

   - Empirically, we demonstrate the efficiency and scalability of the proposed framework on a wide variety of large-scale real-world datasets; one of which contains more than 2 millions data points (Section 5.3).

Chapter 2

Related Works

2.1 Reinforcement Learning (RL)

In reinforcement learning (RL), an agent faces a dilemma between acting optimally with respect to the current, possibly incomplete knowledge of the environment (i.e., exploitation) vs. acting sub-optimally to gain more information about it (i.e., exploration). Model-based Bayesian reinforcement learning (BRL) circumvents such a dilemma by considering the notion of Bayes-optimality [Duff, 2003]: A Bayes-optimal policy selects actions that maximize the agent's expected utility with respect to all possible sequences of future beliefs (starting from the initial belief) over candidate models of the environment. Unfortunately, due to the large belief space, the Bayes-optimal policy can only be approximately derived under a simple choice of models and model priors. For example, the Flat-Dirichlet-Multinomial (FDM) prior [Poupart *et al.*, 2006; Ross *et al.*, 2007; Poupart and Vlassis, 2008] assumes the next-state distributions for each action-state pair to be modeled as independent multinomial distributions with separate Dirichlet priors. Notably, Poupart *et al.* [2006] shows that it is computationally feasible to analytically derive the ex-

act Bayes-optimal policy under the FDM parameterization and proposes practical algorithm to achieve it efficiently. However, despite its common use to analyze and benchmark algorithms, FDM can perform poorly in practice as it often fails to exploit the structured information of a problem [Asmuth and Littman, 2011; A.-Lopez *et al.*, 2012].

To elaborate, a critical limitation of FDM lies in its independence assumption driven by computational convenience rather than scientific insight. We can identify practical examples in the context of self-interested multi-agent RL (MARL) where the uncertainty in the transition model is mainly caused by the stochasticity in the other agent's behavior (in different states) for which the independence assumption does not hold (e.g., motion behavior of pedestrians [Natarajan *et al.*, 2012a; Natarajan *et al.*, 2012b; Natarajan *et al.*, 2014]). Consider, for example, an application of BRL in the problem of placing static sensors to monitor an environmental phenomenon: It involves actively selecting sensor locations (i.e., states) for measurement such that the sum of predictive variances at the unobserved locations is minimized. Here, the phenomenon is the "other agent" and the measurements are its actions. An important characterization of the phenomenon is that of the spatial correlation of measurements between neighboring locations/states [Low *et al.*, 2007; Low *et al.*, 2008; Low *et al.*, 2009; Low *et al.*, 2011; Low *et al.*, 2012; Chen *et al.*, 2012; Cao *et al.*, 2013], which makes FDM-based BRL ill-suited for this problem due to its independence assumption.

Secondly, despite its computational convenience, FDM does not permit generalization across states [Asmuth and Littman, 2011], thus severely limiting its applicability in practical problems with a large state space where past observations only come from a very limited set of states. Interestingly, in such problems, it is often possible to obtain prior domain knowledge providing a more "parsimonious" structure of the

other agent's behavior, which can potentially resolve the issue of generalization. For example, consider using BRL to derive a Bayes-optimal policy for an autonomous car to navigate successfully among human-driven vehicles [Hoang and Low, 2012; Hoang and Low, 2013a; Hoang and Low, 2013b] whose behaviors in different situations (i.e., states) are governed by a small, consistent set of latent parameters, as demonstrated in the empirical study of Gipps [1981]. By estimating/learning these parameters, it is then possible to generalize their behaviors across different states. This, however, contradicts the independence assumption of FDM; in practice, ignoring this results in an inferior performance, as shown in Section 3.3. Note that, by using parameter tying [Poupart *et al.*, 2006], FDM can be modified to make the other agent's behavior identical in different states. But, this simple generalization is too restrictive for real-world problems like the examples above where the other agent's behavior in different states is not necessarily identical but related via a common set of latent "non-Dirichlet" parameters.

Consequently, there is still a huge gap in putting BRL into practice for interacting with self-interested agents of unknown behaviors. To the best of our knowledge, this is first investigated by Chalkiadakis and Boutilier [2003] who offer a myopic solution in the belief space instead of solving for a Bayes-optimal policy that is nonmyopic. Their proposed BPVI method essentially selects actions that jointly maximize a heuristic aggregation of myopic value of perfect information [Dearden *et al.*, 1998] and an average estimation of expected utility obtained from solving the exact MDPs with respect to samples drawn from the posterior belief of the other agent's behavior. Moreover, BPVI is restricted to work only with Dirichlet priors and multinomial likelihoods (i.e., FDM), which are subject to the above disadvantages in modeling the other agent's behavior. Also, BPVI is demonstrated empirically in the simplest of settings with only a few states.

Furthermore, in light of the above examples, the other agent's behavior often needs to be modeled differently depending on the specific application. Grounding in the context of the BRL framework, either the domain expert struggles to best fit his prior knowledge to the supported set of models and model priors or the agent developer has to re-design the framework to incorporate a new modeling scheme. Arguably, there is no free lunch when it comes to modeling the other agent's behavior across various applications. To cope with this difficulty, the BRL framework should ideally allow a domain expert to freely incorporate his choice of design in modeling the other agent's behavior.

In fact, to the best of our knowledge, Monte Carlo BRL (MC-BRL) [Wang *et al.*, 2012] is the only recent work that does not require conjugate distributions to encode prior knowledge: It samples *a priori* a finite set of candidate models to approximately represent the continuous model spectrum and consequently, cast BRL as a discrete POMDP problem, which is relatively easy to solve with point-based approximation algorithms [Pineau *et al.*, 2003; Spaan and Vlassis, 2005; Kurniawati *et al.*, 2008]. That said, using a finite set of candidate models to represent the continuous spectrum of models appears rigid and less robust as it effectively assigns zero probability to the uncovered areas of the spectrum. The performance quality of this approach, therefore, depends on whether the true model is sufficiently similar to the sampled candidates [Wang *et al.*, 2012]. This thesis thus introduces an alternative solution to BRL which also does not require specific parametric modeling to encode the domain expert's prior knowledge and unlike MC-BRL, it does not strictly impose zero probability on models which are not covered by its samples. More interestingly, we show that MC-BRL can also be interpreted as a specific instance of our general framework using a simple choice of basis functions, which are detailed later in Chapter 3.2.3.

Finally, we would like to note that while solving for the Bayes-optimal policy efficiently has not been addressed explicitly in general prior to this thesis, we can actually avoid this problem by allowing the agent to act sub-optimally in a bounded number of steps. In particular, the works of Kolter and Ng [2009], Asmuth and Littman [2011] andA.-Lopez *et al.* [2012] all guarantee that, in the worst case, the agent will act nearly approximately Bayes-optimal in all but a polynomially bounded number of steps with high probability. Alternatively, another approach is to explicitly modify the objective reward function by adding a reward bonus for exploration [Sorg *et al.*, 2010] which also results in similar bounded sample complexity of learning an MDP as of the above algorithms (e.g., [Kolter and Ng, 2009]). It is thus necessary to point out the difference between I-BRL and these worst-case approaches: We are interested in maximizing the average-case performance with certainty rather than the worst-case performance with some "high probability" guarantee. Comparing their performances is beyond the scope of this thesis.

Other non-BRL Works in MARL. In self-interested (or non-cooperative) MARL, there has been several groups of proponents advocating different learning goals, the following of which have garnered substantial support: (a) **Stability** − in self-play or against a certain class of learning opponents, the learners' behaviors converge to an equilibrium; (b) **optimality** − a learner's behavior necessarily converges to the best policy against a certain class of learning opponents; and (c) **security** − a learner's average payoff must exceed the maximin value of the game. For example, the works of Littman [2001], Bianchi *et al.* [2007], and Akchurina [2009] have focused on (evolutionary) game-theoretic approaches that satisfy the **stability** criterion in self-play. The works of Bowling and Veloso [2001], Suematsu and Hayashi [2002], and Tesauro [2003] have developed algorithms that address both the **optimality** and

stability criteria: A learner essentially converges to the best response if the opponents' policies are stationary; otherwise, it converges in self-play. Notably, the work of Powers and Shoham [2005] has proposed an approach that provably converges to an ϵ-best response (i.e., **optimality**) against a class of adaptive, bounded-memory opponents while simultaneously guaranteeing a minimum average payoff (i.e., **security**) in single-state, repeated games.

In contrast to the above-mentioned works that focus on convergence, I-BRL directly optimizes a learner's performance during its course of interaction, which may terminate before it can successfully learn its opponent's behavior. So, our main concern is how well the learner can perform before its behavior converges. From a practical perspective, this seems to be a more appropriate goal: In reality, the agents may only interact for a limited period, which is not enough to guarantee convergence, thus undermining the **stability** and **optimality** criteria. In such a context, the existing approaches appear to be at a disadvantage: (a) Algorithms that focus on **stability** and **optimality** tend to select exploratory actions with drastic effect without considering their huge costs (i.e., poor rewards) [Chalkiadakis and Boutilier, 2003]; and (b) though the notion of **security** aims to prevent a learner from selecting such radical actions, the proposed security values (e.g., maximin value) may not always turn out to be tight lower bounds for the optimal performance.

2.2 Active Learning (AL)

Active learning has become an increasingly important focal theme in many environmental sensing and monitoring applications (e.g., precision agriculture [Tokekar *et al.*, 2013], mineral prospecting [Low *et al.*, 2007], monitoring of ocean and freshwater phenomena like harmful algal blooms [Leonard *et al.*, 2007; Dolan *et al.*, 2009;

Podnar *et al.*, 2010], forest ecosystems, or pollution) where a high-resolution *in situ* sampling of the spatial phenomenon of interest is impractical due to prohibitively costly sampling budget requirements (e.g., number of deployed sensors, energy consumption, mission time): For such applications, it is thus desirable to select and gather the *most informative* observations/data for modeling and predicting the spatially varying phenomenon subject to some budget constraints, which is the goal of active learning and also known as the *active sensing* problem.

To elaborate, solving the active sensing problem amounts to deriving an optimal sequential policy that plans/decides the most informative locations to be observed for minimizing the predictive uncertainty of the unobserved areas of a phenomenon given a sampling budget. To achieve this, many existing active sensing algorithms [Cao *et al.*, 2013; Chen *et al.*, 2012; Chen *et al.*, 2013c; Krause *et al.*, 2008; Low *et al.*, 2008; Low *et al.*, 2009; Low *et al.*, 2011; Low *et al.*, 2012; Singh *et al.*, 2009] have modeled the phenomenon as a *Gaussian process* (GP), which allows its spatial correlation structure to be formally characterized and its predictive uncertainty to be formally quantified (e.g., based on mean-squared error, entropy, or mutual information criterion). However, they have assumed the spatial correlation structure (specifically, the parameters defining it) to be known, which is often violated in real-world applications, or estimated crudely using sparse prior data. So, though they aim to select sampling locations that are optimal with respect to the assumed or estimated parameters, these locations tend to be sub-optimal with respect to the true parameters, thus degrading the predictive performance of the learned GP model.

In practice, the spatial correlation structure of a phenomenon is usually not known. Then, the predictive performance of the GP modeling the phenomenon depends on how informative the gathered observations/data are for both parameter estimation

as well as spatial prediction given the true parameters. Interestingly, as revealed in previous geostatistical studies [Martin, 2001; Müller, 2007], policies that are efficient for parameter estimation are not necessarily efficient for spatial prediction with respect to the true model. Thus, the active sensing problem involves a potential trade-off between sampling the most informative locations for spatial prediction given the current, possibly incomplete knowledge of the model parameters (i.e., exploitation) vs. observing locations that gain more information about the parameters (i.e., exploration):

> How then does an active sensing algorithm trade off between these two possibly conflicting sampling objectives?

To tackle this question, one principled approach is to frame active sensing as a sequential decision problem that jointly and naturally optimizes the above exploration-exploitation trade-off while maintaining a Bayesian belief over the model parameters. This intuitively means a policy that biases towards observing informative locations for spatial prediction given the current model prior may be penalized if it entails a highly dispersed posterior over the model parameters. So, the resulting induced policy is guaranteed to be optimal in the expected active sensing performance. Unfortunately, such a nonmyopic Bayes-optimal policy cannot be derived exactly due to an uncountable set of candidate observations and unknown model parameters [Solomon and Zacks, 1970]. As a result, most existing works [Diggle, 2006; Houlsby *et al.*, 2012; Park and Pillow, 2012; Zimmerman, 2006; Ouyang *et al.*, 2014] have circumvented the trade-off by resorting to the use of myopic/greedy (hence, sub-optimal) policies.

To the best of our knowledge, the only notable nonmyopic active sensing algorithm for GPs [Krause and Guestrin, 2007] advocates tackling exploration and exploitation separately, instead of jointly and naturally optimizing their trade-off, to sidestep the

difficulty of solving the Bayesian sequential decision problem. Specifically, it performs a probably approximately correct (PAC)-style exploration until it can verify that the performance loss of greedy exploitation lies within a user-specified threshold. But, such an algorithm is sub-optimal in the presence of budget constraints due to the following limitations: (a) It is unclear how an optimal threshold for exploration can be determined given a sampling budget, and (b) even if such a threshold is available, the PAC-style exploration is typically designed to satisfy a worst-case sample complexity rather than to be optimal in the expected active sensing performance, thus resulting in an overly-aggressive exploration (Section 4.3.1). Notably, Cuong *et al.* [2014] have recently introduced alternative AL criteria for which there exists greedy strategies that achieve a constant factor approximation to the corresponding optimal policy. This approach, however, focuses on the space of parametric models which tacitly assume that given the true model, the probability of getting a particular observation at a previously unseen location does not depend on the collected data, thus avoiding the infamous exploration-exploitation trade-off between sampling for spatial prediction and parameter estimation. In addition, this work is also grounded in the context of classification for which the set of candidate observations is finite. In contrast, our work in this thesis does not assume that the set of candidate observations is finite and more importantly, we directly address this active sensing problem in the context of non-parametric model space (Chapter 4).

On a different avenue of development, there also exists other lines of works [Hanneke, 2007; Balcan *et al.*, 2009; Golovin *et al.*, 2010; Yang *et al.*, 2011; Dasgupta *et al.*, 2007; Beygelzimer *et al.*, 2009] in the AL literature which advocate the use of greedy algorithms whose rates of convergence (i.e., the number of samples (experiments) required for the learner to achieve a desired performance in the worst case) are provably minimax optimal with respect to some simple classes of learning problems (e.g., binary

classification using parametric hypothesis spaces with finite VC dimensions, assuming independently and identically distributed observations, etc.). However, these works do not explicitly consider the trade-off between exploration and exploitation in the presence of budget constraints as well as problem domains with far more complicated structures. Thus, despite the milestone contributions they have made in terms of the convergence rate in active learning, significantly less is presently known about how well these proposed algorithms balance between exploration and exploitation given a fixed budget for interaction: Specifically, how good are their expected performance in practical, complex environmental domains, and more importantly, whether it is possible to derive a trade off between exploration and exploitation that achieves the optimal expected performance in such sophisticated environments? In fact, we find that, unlike its counterpart in RL, the exploration-exploitation trade-off in AL has not been received much attention from the research community until recently [Krause and Guestrin, 2007] and is still a research topic in its infancy.

2.3 Sparse Gaussian Process-Based Learning Models for Big Data

The 21st century marks the beginning of the big data era in which we are facing the problem of scalability. Existing machine learning (ML) models which are developed in the previous decades can no longer cope up with the prohibitively expensive cost of processing massive datasets. As a striking example, while Gaussian Process (GP) [Rasmussen and Williams, 2006] appears to be one of the most competitive approaches for Bayesian non-parametric regression, it incurs $\mathcal{O}(n^3)$ processing time for datasets of size n. This highly expensive computational cost thus effectively renders GP completely useless in handling modern time datasets which may contain millions

of data points.

To overcome this computational disadvantage, a wealth of sparse GP (SGP) regression methods [Quiñonero-Candela and Rasmussen, 2005; Snelson and Ghahramani, 2007; Titsias, 2009; Lázaro-Gredilla *et al.*, 2010] have been proposed and developed by numerous authors in the past few years. A common trait to many of these approaches is the assumption of conditional independence between different blocks of latent variables given a separate, small subset of m inducing latent variables which are distributed by the same GP: The resulting models are then able to offer a reduced computational complexity of $\mathcal{O}(nm^2)$. In fact, this appears to be the main recipe to derive a class of well-known SGP models [Quiñonero-Candela and Rasmussen, 2005] which include Subset of Regressors (SoR) [Smola and Bartlett, 2001], Deterministic Training Conditional (DTC) [Seeger *et al.*, 2003], Partially Independent Training Conditional (PITC) [Schwaighofer and Tresp, 2003] and Fully Independent Training Conditional (FITC) [Snelson and Gharahmani, 2006] as well as their improved variants Fully Independent Conditional (FIC) and Partially Independent Conditional (PIC) [Snelson and Ghahramani, 2007]. Remarkably, Chen *et al.* [2013a] successfully exploit the low-rank covariance matrix approximation of FI(T)C and PI(T)C [Snelson and Ghahramani, 2007] to introduce a framework of parallel SGPs which distributes its computational load among parallel machines to achieve better scalability.

Unfortunately, even these SGP models are impractical for big data as their reduced computation cost $\mathcal{O}(nm^2)$ only scales up to middle-size datasets with only tens of thousand data points[1]. To the best of our knowledge, the only existing work capable of processing millions of data points has been recently introduced in [Hensman

[1] As a matter of fact, the parallel SGPs [Chen *et al.*, 2013b] are evaluated with datasets containing less than $50,000$ data points.

et al., 2013] which provides an anytime version of DTC for big data. In particular, Hensman *et al.* [2013] exploit the fact that DTC can be derived by minimizing the KL-divergence [Titsias, 2009] between its approximated posterior and the exact GP posterior over latent variables. This interestingly reveals an alternative numerical computation process via *stochastic gradient ascent* (SGA) which asymptotically converges towards DTC and only incurs $\mathcal{O}(m^3)$ processing time per iteration. The proposed approach thus promises a remarkable speed-up if the number of iterations required for convergence is significantly smaller than n.

This approach, however, focuses on faithfully converging towards DTC rather than preserving the current state-of-the-art performance of SGP on big data. In fact, the choice of DTC appears superficially imposed so that one can take advantage of its readily available SGA-based numerical computation whose complexity per iteration is independent of n. In terms of predictive performance, PIC [Snelson, 2007] can be regarded as the current state-of-the-art which, as a matter of fact, is shown to consistently outperform DTC on a wide range of datasets (Section 5.3). This is expected because unlike DTC, PIC does not forcibly assume a deterministic relation between the inducing variables and others which appears to be an overly strong assumption. Furthermore, according to our experiments in Section 5.3, the anytime version of DTC [Hensman *et al.*, 2013] always performs significantly worse than that of PIC and in addition, less competitive to PITC during the early stage of the anytime approximation. Interestingly, in terms of the predictive variance, Snelson and Ghahramani [2007] previously demonstrated that DTC's prediction catastrophically *breaks* at locations near the inducing point: Its prediction wrongly deviates from the exact measurements at these locations yet its variance is almost close to zero (e.g., high confidence) due to its deterministic assumption[2]. Thus, when using as

[2]For more details, please refer to Chapter 2.3.8 of [Snelson and Ghahramani, 2007].

the underlying predictive model for AL algorithms where it is crucially important to have a good estimation of the predictive variance (Chapter 4), this over-confident behavior of DTC appears to be harmfully misleading. This essentially boils down to the question of whether it is possible to construct a similar SGA-based numerical process which is both computationally efficient and convergent towards a particular SGP model of our choice since depending on particular situations, one SGP model might perform better than the others and vice versa. Unfortunately, the alternative numerical computation processes of the other SGPs, unlike DTC's, are not readily available from their derivations which are not based on optimization.

Chapter 3

Interactive Bayesian Reinforcement Learning (I-BRL)

Motivated by the practical considerations in Section 2.1, this chapter presents a novel generalization of BRL, which we call *Interactive BRL* (I-BRL) (Section 3.2), to integrate any parametric model and model prior of the other agent's behavior (Section 3.1) specified by domain experts, consequently yielding two advantages: The other agent's behavior can be represented (a) in a fine-grained manner based on the practitioners' prior domain knowledge, and (b) compactly to be generalized across different states, thus overcoming the limitations of FDM. We show how the non-myopic Bayes-optimal policy can be derived analytically by solving I-BRL exactly (Section 3.2.1) and propose an approximation algorithm to compute it efficiently in polynomial time (Section 3.2.2). Empirically, we evaluate the performance of I-BRL against that of BPVI [Chalkiadakis and Boutilier, 2003] and MC-BRL [Wang *et al.*, 2012] using an interesting traffic problem modeled after a real-world situation (Section 3.3.2). Although I-BRL tailors towards multi-agent settings, the developed theory is also applicable to single-agent RL by treating the environment as the other agent.

3.1 Modeling the Other Agent

In our proposed Bayesian modeling paradigm, the opponent's[1] behavior is modeled as a set of probabilities $p^v_{sh}(\lambda) \triangleq \Pr(v|s,h,\lambda)$ for selecting action v in state s conditioned on the history $h \triangleq \{s_i, u_i, v_i\}_{i=1}^{d}$ of d latest interactions where u_i is the action taken by our agent in the i-th step. These distributions are parameterized by λ, which abstracts the actual parametric form of the opponent's behavior; this abstraction provides practitioners the flexibility in choosing the most suitable degree of parameterization. For example, λ can simply be a set of multinomial distributions $\lambda \triangleq \{\theta^v_{sh}\}$ such that $p^v_{sh}(\lambda) \triangleq \theta^v_{sh}$ if no prior domain knowledge is available. Otherwise, the domain knowledge can be exploited to produce a fine-grained representation of λ; at the same time, λ can be made compact to generalize the opponent's behavior across different states (e.g., Section 3.3).

The opponent's behavior can be learned by monitoring the belief $b(\lambda) \triangleq \Pr(\lambda)$ over all possible λ. In particular, the belief (or probability density) $b(\lambda)$ is updated at each step based on the history $h \circ \langle s,u,v \rangle$ of $d+1$ latest interactions (with $\langle s,u,v \rangle$ being the most recent one) using Bayes' theorem:

$$b^v_{sh}(\lambda) \;\; \propto \;\; p^v_{sh}(\lambda)\; b(\lambda)\;. \tag{3.1}$$

Let $\bar{s} = (s,h)$ denote an information state that consists of the current state and the history of d latest interactions. When the opponent's behavior is stationary (i.e., $d = 0$), it follows that $\bar{s} = s$. For ease of notations, the main results of our work (in subsequent sections) are presented only for the case where $d = 0$ (i.e., $\bar{s} = s$); extension to the general case just requires replacing s with $\bar{s}$. In this case, (3.1) can

[1]For convenience, we will use the terms the "other agent" and "opponent" interchangeably from throughout this chapter.

be re-written as

$$b_s^v(\lambda) \quad \propto \quad p_s^v(\lambda)\ b(\lambda)\ . \tag{3.2}$$

The key difference between our Bayesian modeling paradigm and FDM [Poupart *et al.*, 2006] is that we do not require $b(\lambda)$ and $p_s^v(\lambda)$ to be, respectively, Dirichlet prior and multinomial likelihood where Dirichlet is a conjugate prior for multinomial. In practice, such a conjugate prior is desirable because the posterior b_s^v belongs to the same Dirichlet family as the prior b, thus making the belief update tractable and the Bayes-optimal policy efficient to be derived. Despite its computational convenience, this conjugate prior restricts the practitioners from exploiting their domain knowledge to design more informed priors (e.g., see Section 3.3). Furthermore, this turns out to be an overkill just to make the belief update tractable. In particular, we show in Theorem 1 below that, without assuming any specific parametric form of the initial prior, the posterior belief can still be tractably represented even though they are not necessarily conjugate distributions. This is indeed sufficient to guarantee and derive a tractable representation of the Bayes-optimal policy using a finite set of parameters, as shall be seen later in Section 3.2.1.

Theorem 1. *If the initial prior b can be represented exactly using a finite set of parameters, then the posterior b' conditioned on a sequence of observations $\{(s_i, v_i)\}_{i=1}^{n'}$ can also be represented exactly in parametric form. This is achievable, as detailed in the below proof sketch, because b' only depends on certain statistics of $\{(s_i, v_i)\}_{i=1}^{n'}$, whose storage complexity is independent of n', instead of the entire sequence itself.*

Proof Sketch. From (3.2), we can prove by induction on n' that

$$b'(\lambda) \propto \Phi(\lambda)b(\lambda) \tag{3.3}$$

$$\Phi(\lambda) \triangleq \prod_{s\in S}\prod_{v\in V} p_s^v(\lambda)^{\psi_s^v} , \tag{3.4}$$

where $\psi_s^v \triangleq \sum_{i=1}^{n'} \delta_{sv}(s_i, v_i)$ and δ_{sv} is the Kronecker delta function that returns 1 if $s = s_i$ and $v = v_i$, and 0 otherwise[2]. From (3.3), it is clear that b' can be represented by a set of parameters $\{\psi_s^v\}_{s,v}$ and the finite representation of b. Thus, belief update is performed simply by incrementing the hyper-parameter ψ_s^v according to each observation (s, v). □

3.2 Interactive Bayesian Reinforcement Learning

In this section, we first extend the proof techniques used in [Poupart *et al.*, 2006] to theoretically derive the agent's Bayes-optimal policy against the general class of parametric models and model priors of the opponent's behavior (Section 3.1). In particular, we show that the derived Bayes-optimal policy can also be represented exactly using a finite number of parameters. Based on our derivation, a naive algorithm can be devised to compute the exact parametric form of the Bayes-optimal policy (Section 3.2.1). Finally, we present a practical algorithm to efficiently approximate this Bayes-optimal policy in polynomial time (with respect to the size of the environment model) (Section 3.2.2).

Formally, an agent is assumed to be interacting with its opponent in a stochastic environment modeled as a tuple $(S, U, V, \{r_s\}, \{p_s^{uv}\}, \{p_s^v(\lambda)\}, \phi)$ where S is a finite

[2]Intuitively, $\Phi(\lambda)$ can be interpreted as the likelihood of observing each pair (s, v) for ψ_s^v times while interacting with an opponent whose behavior is parameterized by λ.

set of states, U and V are sets of actions available to the agent and its opponent, respectively. In each stage, the immediate payoff $r_s(u, v)$ to our agent depends on the joint action $(u, v) \in U \times V$ and the current state $s \in S$. The environment then transitions to a new state s' with probability $p_s^{uv}(s') \triangleq \Pr(s'|s, u, v)$ and the future payoff (in state s') is discounted by a constant factor $0 < \phi < 1$, and so on. Finally, as described in Section 3.1, the opponent's latent behavior $\{p_s^v(\lambda)\}$ can be selected from the general class of parametric models and model priors, which subsumes FDM (i.e., independent multinomials with separate Dirichlet priors).

Now, let us recall that the key idea underlying the notion of Bayes-optimality [Duff, 2003] is to maintain a belief $b(\lambda)$ that represents the uncertainty surrounding the opponent's behavior λ in each stage of interaction. Thus, the action selected by the learner in each stage affects both its expected immediate payoff $\mathbb{E}_\lambda[\sum_v p_s^v(\lambda) r_s(u, v)|b]$ and the posterior belief state $b_s^v(\lambda)$, the latter of which influences its future payoff and builds in the information gathering option (i.e., exploration). As such, the Bayes-optimal policy can be obtained by maximizing the expected discounted sum of rewards $V_s(b)$ as detailed below:

$$V_s(b) \triangleq \max_u \sum_v \langle p_s^v, b \rangle \left(r_s(u, v) + \phi \sum_{s'} p_s^{uv}(s') V_{s'}(b_s^v) \right) , \tag{3.5}$$

where $\langle a, b \rangle \triangleq \int_\lambda a(\lambda) b(\lambda) \mathrm{d}\lambda$. The optimal policy for the learner is then defined as a function π^* that maps the belief b to an action u maximizing its expected utility, which can be derived by solving (3.5). To derive our solution, we first re-state two well-known results concerning the augmented belief-state MDP in single-agent RL [Poupart *et al.*, 2006], which also hold straight-forwardly for our general class of parametric models and model priors.

Theorem 2. *The optimal value function V^k for k steps-to-go converges to the optimal value function V for infinite horizon as $k \to \infty$:*

$$\|V - V^{k+1}\|_\infty \;\; \leq \;\; \phi \|V - V^k\|_\infty \; . \tag{3.6}$$

Theorem 3. *The optimal value function $V_s^k(b)$ for k steps-to-go can be represented by a finite set Γ_s^k of α-functions:*

$$V_s^k(b) \;\; = \;\; \max_{\alpha_s \in \Gamma_s^k} \langle \alpha_s, b \rangle \; . \tag{3.7}$$

Simply put, these results imply that the optimal value V_s in (3.5) can be approximated arbitrarily closely by a finite set Γ_s^k of piecewise linear α-functions α_s, as shown in (3.7). Each α-function α_s is associated with an action u_{α_s} yielding an expected utility of $\alpha_s(\lambda)$ if the true behavior of the opponent is λ and consequently an overall expected reward $\langle \alpha_s, b \rangle$ by assuming that, starting from (s, b), the learner selects action u_{α_s} and continues optimally thereafter. In particular, Γ_s^k and u_{α_s} can be derived based on a constructive proof of Theorem 3. However, for the sake of clarity, we only state the constructive process below. Interested readers are referred to Appendix A for a detailed proof. Specifically, given $\{\Gamma_s^k\}_s$ such that (3.7) holds for k, it follows (see Appendix A) that

$$V_s^{k+1}(b) \;\; = \;\; \max_{u,t} \left\langle \alpha_s^{ut}, b \right\rangle \; , \tag{3.8}$$

where $t \stackrel{\triangle}{=} (t_{s'v})_{s' \in S, v \in V}$ with $t_{s'v} \in \left\{1, \ldots, \left|\Gamma_{s'}^k\right|\right\}$, and

$$\alpha_s^{ut}(\lambda) \;\; \stackrel{\triangle}{=} \;\; \sum_v p_s^v(\lambda) \Big(r_s(u, v) + \phi \sum_{s'} \alpha_{s'}^{t_{s'v}}(\lambda) p_s^{uv}(s') \Big) \; , \tag{3.9}$$

such that $\alpha_{s'}^{t_{s'v}}$ denotes the $t_{s'v}$-th α-function in $\Gamma_{s'}^k$. Setting $\Gamma_s^{k+1} = \{\alpha_s^{ut}\}_{u,t}$ and

$u_{\alpha_s^{ut}} = u$, it follows that (3.7) also holds for $k+1$. As a result, the optimal policy $\pi^*(b)$ can be derived directly from these α-functions by $\pi^*(b) \triangleq u_{\alpha_s^*}$ where $\alpha_s^* = \arg\max_{\alpha_s^{ut} \in \Gamma_s^{k+1}} \langle \alpha_s^{ut}, b \rangle$. Thus, constructing Γ_s^{k+1} from the previously constructed sets $\{\Gamma_s^k\}_s$ essentially boils down to an exhaustive enumeration of all possible pairs (u,t) and the corresponding application of (3.9) to compute α_s^{ut}. Though (3.9) specifies a bottom-up procedure constructing Γ_s^{k+1} from the previously constructed sets $\{\Gamma_{s'}^k\}_{s'}$ of α-functions, it implicitly requires a convenient parameterization for the α-functions that is closed under the application of (3.9). To complete this analytical derivation, we present a final result to demonstrate that each α-function is indeed of such parametric form. Note that Theorem 4 below generalizes a similar result proven in [Poupart *et al.*, 2006], the latter of which shows that, under FDM, each α-function can be represented by a linear combination of multivariate monomials. A practical algorithm building on our generalized result in Theorem 4 is presented in Section 3.2.2.

Theorem 4. *Let $\mathbf{\Phi}$ denote a family of all functions $\Phi(\lambda)$ (3.4). Then, the optimal value $V_{s'}^k$ can be represented by a finite set $\Gamma_{s'}^k$ of α-functions $\alpha_{s'}^j$ for $j = 1, \ldots, |\Gamma_{s'}^k|$:*

$$\alpha_{s'}^j(\lambda) \;=\; \sum_{i=1}^{m} c_i \Phi_i(\lambda) \;, \tag{3.10}$$

where $\Phi_i \in \mathbf{\Phi}$. So, each α-function $\alpha_{s'}^j$ can be compactly represented by a finite set of parameters $\{c_i\}_{i=1}^m$[3].

Proof Sketch. We will prove (3.10) by induction on k[4]. Supposing (3.10) holds for

[3]To ease readability, we abuse the notations $\{c_i, \Phi_i\}_{i=1}^m$ slightly: Each $\alpha_{s'}^j(\lambda)$ should be specified by a different set $\{c_i, \Phi_i\}_{i=1}^m$.

[4]When $k = 0$, (3.10) can be verified by letting $c_i = 0$.

k. Setting $j = t_{s'v}$ in (3.10) results in

$$\alpha_{s'}^{t_{s'v}}(\lambda) \;=\; \sum_{i=1}^{m} c_i \Phi_i(\lambda) \;, \tag{3.11}$$

which is then plugged into (3.9) to yield

$$\alpha_s^{ut}(\lambda) \;=\; \sum_{v \in V} c_v \Psi_v(\lambda) + \sum_{s' \in S} \sum_{v \in V} \left(\sum_{i=1}^{m} c_{s'i}^v \Psi_{s'i}^v(\lambda) \right) \;, \tag{3.12}$$

where $\Psi_v(\lambda) = p_s^v(\lambda)$, $\Psi_{s'i}^v(\lambda) = p_s^v(\lambda)\Phi_i(\lambda)$, and the coefficients $c_v = r_s(u, v)$ and $c_{s'i}^v = \phi p_s^{uv}(s')c_i$. It is easy to see that $\Psi_v \in \mathbf{\Phi}$ and $\Psi_{s'i}^v \in \mathbf{\Phi}$. So, (3.10) clearly holds for $k+1$. We have shown above that, under the general class of parametric models and model priors (Section 3.1), each α-function can be represented by a linear combination of arbitrary parametric functions in $\mathbf{\Phi}$, which subsume multivariate monomials used in [Poupart *et al.*, 2006]. $\square$

3.2.1 An Exact Algorithm

Intuitively, Theorems 3 and 4 provide a simple and constructive method for computing the set of α-functions and hence, the optimal policy. In step $k+1$, the sets Γ_s^{k+1} for all $s \in S$ are constructed using (3.11) and (3.12) from $\Gamma_{s'}^k$ for all $s' \in S$, the latter of which are computed previously in step k. When $k = 0$ (i.e., base case), see the proof of Theorem 4 above (i.e., footnote 4). A sketch of this algorithm is shown below:

BACKUP$(s, k+1)$

1. $\Gamma_{s,u}^* \leftarrow \left\{ g(\lambda) \triangleq \sum_v c_v \Psi_v(\lambda) \right\}$
2. $\Gamma_{s,u}^{v,s'} \leftarrow \left\{ g_j(\lambda) \triangleq \sum_{i=1}^{m} c_{s'i}^v \Psi_{s'i}^v(\lambda) \right\}_{j=1,\ldots,|\Gamma_{s'}^k|}$

3. $\Gamma_{s,u} \leftarrow \Gamma^{*}_{s,u} \oplus \left(\bigoplus_{v,s'} \Gamma^{v,s'}_{s,u} \right)$ [5]

4. $\Gamma^{k+1}_{s} \leftarrow \bigcup_{u \in U} \Gamma_{s,u}$

In the above algorithm, steps **1** and **2** compute the first and second summation terms on the right-hand side of (3.12), respectively. Then, steps **3** and **4** construct $\Gamma^{k+1}_{s} = \{\alpha^{ut}_{s}\}_{u,t}$ using (3.12) over all t and u, respectively. Thus, by iteratively computing $\Gamma^{k+1}_{s} = \mathbf{BACKUP}(s, k+1)$ for a sufficiently large value of k, Γ^{k+1}_{s} can be used to approximate V_s arbitrarily closely, as shown in Theorem 2. However, this naive algorithm is computationally impractical due to the following issues: (a) **α-function explosion** − the number of α-functions grows doubly exponentially in the planning horizon length, as derived from (3.8) and (3.9): $\left|\Gamma^{k+1}_{s}\right| = \mathcal{O}\left(\left[\prod_{s'} \left|\Gamma^{k}_{s'}\right|\right]^{|V|} |U|\right)$, and (b) **parameter explosion** − the average number of parameters used to represent an α-function grows by a factor of $\mathcal{O}(|S||V|)$, as manifested in (3.12). The practicality of our approach therefore depends crucially on how these issues are resolved, as described in the next section.

3.2.2 A Practical Approximation Algorithm

In this section, we introduce practical modifications of the **BACKUP** algorithm by addressing the above-mentioned issues. We first address the issue of **α-function explosion** by generalizing discrete POMDP's PBVI solver [Pineau *et al.*, 2003] to be used for our augmented belief-state MDP: Only the α-functions that yield optimal values for a sampled set of reachable beliefs $B_s = \{b^1_s, b^2_s, \cdots, b^{|B_s|}_s\}$ are computed (see the modifications in steps **3** and **4** of the **PB-BACKUP** algorithm). The resulting algorithm is shown below:

[5] $A \oplus B = \{a + b | a \in A, b \in B\}$.

PB-BACKUP$(B_s = \{b_s^1, b_s^2, \cdots, b_s^{|B_s|}\}, s, k+1)$

1. $\Gamma_{s,u}^{*} \leftarrow \left\{ g(\lambda) \triangleq \sum_{v} c_v \Psi_v(\lambda) \right\}$
2. $\Gamma_{s,u}^{v,s'} \leftarrow \left\{ g_j(\lambda) \triangleq \sum_{i=1}^{m} c_{s'i}^{v} \Psi_{s'i}^{v}(\lambda) \right\}_{j=1,\ldots,|\Gamma_{s'}^{k}|}$
3. $\Gamma_{s,u}^{i} \leftarrow \left\{ g + \sum_{s',v} \arg\max_{g_j \in \Gamma_{s,u}^{v,s'}} \left\langle g_j, b_s^i \right\rangle \right\}_{g \in \Gamma_{s,u}^{*}}$
4. $\Gamma_{s}^{k+1} \leftarrow \left\{ g_i \triangleq \arg\max_{g \in \Gamma_{s,u}^{i}} \left\langle g, b_s^i \right\rangle \right\}_{i=1,\ldots,|B_s|}$

Secondly, to address the issue of **parameter explosion**, each α-function is projected onto a fixed number of basis functions to keep the number of parameters from growing exponentially. This projection is done after each **PB-BACKUP** operation, hence always keeping the number of parameters fixed (i.e., one parameter per basis function). In particular, since each α-function is in fact a linear combination of functions in $\mathbf{\Phi}$ (Theorem 4), it is natural to choose these basis functions from $\mathbf{\Phi}$ (See Section 3.2.3 for other choices). Besides, it is easy to see from (3.3) that each sampled belief b_s^i can also be written as

$$b_s^i(\lambda) \;\; = \;\; \eta \Phi_s^i(\lambda) b(\lambda) \;, \tag{3.13}$$

where b is the initial prior belief, $\eta = 1/\langle \Phi_s^i, b \rangle$, and $\Phi_s^i \in \mathbf{\Phi}$. For convenience, these $\{\Phi_s^i\}_{i=1,\ldots,|B_s|}$ are selected as basis functions. Specifically, after each **PB-BACKUP** operation, each $\alpha_s \in \Gamma_s^k$ is projected onto the function space defined by $\{\Phi_s^i\}_{i=1,\ldots,|B_s|}$. This projection is then cast as an optimization problem that minimizes the squared difference $J(\alpha_s)$ between the α-function and its projection with respect to the sampled

beliefs in B_s:

$$J(\alpha_s) \triangleq \frac{1}{2}\sum_{j=1}^{|B_s|}\left(\left\langle \alpha_s, b_s^j\right\rangle - \sum_{i=1}^{|B_s|} c_i \left\langle \Phi_s^i, b_s^j\right\rangle\right)^2 . \tag{3.14}$$

This can be done analytically by letting $\frac{\partial J(\alpha_s)}{\partial c_i} = 0$ and solving for c_i, which is equivalent to solving a linear system $Ax = d$ where $x_i = c_i$, $A_{ji} = \sum_{k=1}^{|B_s|}\left\langle \Phi_s^i, b_s^k\right\rangle\left\langle \Phi_s^j, b_s^k\right\rangle$ and $d_j = \sum_{k=1}^{|B_s|}\left\langle \Phi_s^j, b_s^k\right\rangle\left\langle \alpha_s, b_s^k\right\rangle$. Note that this projection works directly with the values $\langle \alpha_s, b_s^j\rangle$ instead of the exact parametric form of α_s in (3.10). This allows for a more compact implementation of the **PB-BACKUP** algorithm presented above: Instead of maintaining the exact parameters that represent each of the immediate functions g, only their evaluations at the sampled beliefs $B_s = \left\{b_s^1, b_s^2, \cdots, b_s^{|B_s|}\right\}$ need to be maintained. In particular, the values of $\{\langle g, b_s^i\rangle\}_{i=1,\ldots,|B_s|}$ can be estimated as follows:

$$\begin{aligned}\left\langle g, b_s^i\right\rangle &= \eta\int_\lambda g(\lambda)\Phi_s^i(\lambda)b(\lambda)\mathrm{d}\lambda \\ &\simeq \frac{\sum_{j=1}^n g(\lambda^j)\Phi_s^i(\lambda^j)}{\sum_{j=1}^n \Phi_s^i(\lambda^j)} ,\end{aligned} \tag{3.15}$$

where $\{\lambda^j\}_{j=1}^n$ are samples drawn from the initial prior b. During the online execution phase, (3.15) is also used to compute the expected payoff for the α-functions evaluated at the current belief $b'(\lambda) = \eta\Phi(\lambda)b(\lambda)$:

$$\langle \alpha_s, b'\rangle \simeq \frac{\sum_{j=1}^n \Phi(\lambda^j)\sum_{i=1}^{|B_s|} c_i\Phi_s^i(\lambda^j)}{\sum_{j=1}^n \Phi(\lambda^j)} . \tag{3.16}$$

So, the real-time processing cost of evaluating each α-function's expected reward at a particular belief is $\mathcal{O}(|B_s|n)$. Since the sampling of $\{b_s^i\}$, $\{\lambda^j\}$ and the computation of $\left\{\sum_{i=1}^{|B_s|} c_i\Phi_s^i(\lambda^j)\right\}$ can be performed in advance, this $\mathcal{O}(|B_s|n)$ cost is

further reduced to $\mathcal{O}(n)$, which makes the action selection incur $\mathcal{O}(|B_s|n)$ cost in total. This is significantly cheaper as compared to the total cost $\mathcal{O}(nk|S|^2|U||V|)$ of online sampling and re-estimating V_s incurred by BPVI [Chalkiadakis and Boutilier, 2003]. Furthermore, note that since the offline computational costs in steps **1** to **4** of **PB-BACKUP**$(B_s, s, k+1)$ and the projection cost, which is cast as the cost of solving a system of linear equations, are always polynomial functions of the interested variables (e.g., $|S|, |U|, |V|, n, |B_s|$), the optimal policy can be approximated in polynomial time.

In addition, Eqs. (3.15) and (3.16) also reveal the difference between MC-BRL [Wang *et al.*, 2012] and I-BRL. While MC-BRL advocates using the sampled models to approximately represent the continuous model spectrum (Chapter 2.1), I-BRL instead uses these models to approximate the evaluation of the basis functions. I-BRL therefore appears to be less rigid than MC-BRL in approximating the exact posterior belief: Unlike MC-BRL which simply assigns zero probability to models that do not belong to the sample set $\{\lambda^j\}_{j=1}^n$, I-BRL instead uses the samples to estimate the probability density for those candidate models λ[6]

$$b'(\lambda) \;=\; \frac{\Phi(\lambda)b(\lambda)}{\int_{\lambda'}\Phi(\lambda')b(\lambda')\mathrm{d}\lambda'} \;\simeq\; \frac{\Phi(\lambda)b(\lambda)}{\frac{1}{n}\sum_{j=1}^{n}\Phi(\lambda^j)}\,, \tag{3.17}$$

and hence, distributes its belief confidence over the model spectrum more flexibly. As a matter of fact, Eq. (3.19) does not assign zero probability to uncovered areas of the model spectrum. This flexibility in fact helps I-BRL to behave more cautious than MC-BRL in complicated and adverse situations, thus achieving better performance (Section 3.3).

[6]This is implicitly absorbed in the approximated evaluation of Eqs. (3.15) and (3.16) above.

3.2.3 Alternative Choice of Basis Functions

This section demonstrates another theoretical advantage of our framework: The flexibility to customize the general point-based algorithm presented in Section 3.2.2 into more manageable forms (e.g., simple, easy to implement, etc.) with respect to different choices of basis functions. Interestingly, these customizations often allow the practitioners to trade off effectively between the performance and sophistication of the implemented algorithm: A simple choice of basis functions may (though not necessarily) reduce its performance but, in exchange, bestows upon it a customization that is more computationally efficient and easier to implement. This is especially useful in practical situations where finding a good enough solution quickly is more important than looking for better yet time-consuming solutions.

As an example, we present such an alternative of the basis functions which conceptually (and interestingly) cast I-BRL as MC-BRL [Wang *et al.*, 2012]. In particular, let $\{\lambda^i\}_{i=1}^n$ be a set of the opponent's models sampled from the initial belief b. Also, let $\Psi_i(\lambda)$ denote a function that returns 1 if $\lambda = \lambda^i$, and 0 otherwise. According to Section 3.2.2, to keep the number of parameters from growing exponentially, we project each α-function onto $\{\Psi_i(\lambda)\}_{i=1}^n$ by minimizing (3.14) or alternatively, the unconstrained squared difference between the α-function and its projection:

$$J(\alpha_s) \;=\; \frac{1}{2}\int_{\lambda}\left(\alpha_s(\lambda) - \sum_{i=1}^{|B_s|} c_i \Phi_s^i(\lambda)\right)^2 \mathrm{d}\lambda \;. \tag{3.18}$$

Now, let us consider (3.9), which specifies the exact solution for (3.5) in Section 3.2. Assume that $\alpha_{s'}^{t_{s'v}}(\lambda)$ is projected onto $\{\Psi_i(\lambda)\}_{i=1}^n$ by minimizing (3.18):

$$\bar{\alpha}_{s'}^{t_{s'v}}(\lambda) \;=\; \sum_{i=1}^{n} \Psi_i(\lambda)\varphi_{s'}^{t_{s'v}}(i) \;. \tag{3.19}$$

where $\{\varphi_{s'}^{t_{s'v}}(i)\}_i$ are the projection coefficients. According to the general point-based algorithm in Section 3.2.2, $\alpha_s^{ut}(\lambda)$ is first computed by replacing $\alpha_{s'}^{t_{s'v}}(\lambda)$ with $\bar{\alpha}_{s'}^{t_{s'v}}(\lambda)$ in equation (3.9):

$$\alpha_s^{ut}(\lambda) = \sum_v p_s^v(\lambda) \left(r_s(u,v) + \phi \sum_{s'} \bar{\alpha}_{s'}^{t_{s'v}}(\lambda) p_s^{uv}(s') \right) . \tag{3.20}$$

Then, following (3.18), $\alpha_s^{ut}(\lambda)$ is projected onto $\{\Psi_i(\lambda)\}_i$ by solving for $\{\varphi_s^{ut}(i)\}_i$ that minimize

$$J(\alpha_s^{ut}) = \frac{1}{2} \int_\lambda \left(\alpha_s^{ut}(\lambda) - \sum_{i=1}^n \Psi_i(\lambda) \varphi_s^{ut}(i) \right)^2 \mathrm{d}\lambda . \tag{3.21}$$

The back-up operation is therefore cast as finding $\{\varphi_s^{ut}(i)\}_i$ that minimize (3.21). To do this, let us define

$$L(\lambda) = \frac{1}{2} \left(\alpha_s^{ut}(\lambda) - \sum_{i=1}^n \Psi_i(\lambda) \varphi_s^{ut}(i) \right)^2$$

and take the corresponding partial derivatives of $L(\lambda)$ with respect to $\{\varphi_s^{ut}(j)\}_j$:

$$\frac{\partial L(\lambda)}{\partial \varphi_s^{ut}(j)} = - \left(\alpha_s^{ut}(\lambda) - \sum_{i=1}^n \Psi_i(\lambda) \varphi_s^{ut}(i) \right) \Psi_j(\lambda) . \tag{3.22}$$

From the definition of $\Psi_j(\lambda)$, it is clear that when $\lambda \neq \lambda^j$, $\dfrac{\partial L(\lambda)}{\partial \varphi_s^{ut}(j)} = 0$. Otherwise, this only happens when

$$\begin{aligned} \alpha_s^{ut}(\lambda^j) &= \sum_{i=1}^n \Psi_i(\lambda^j) \varphi_s^{ut}(i) \\ &= \varphi_s^{ut}(j) \text{ (by def. of } \Psi_i(\lambda)) . \end{aligned} \tag{3.23}$$

On the other hand, by plugging (3.19) into (3.20) and using $\Psi_i(\lambda) = 0 \; \forall \lambda \neq \lambda^i$, $\alpha_s^{ut}(\lambda^j)$ can be expressed as

$$\alpha_s^{ut}(\lambda^j) \;=\; \sum_v p_s^v(\lambda^j) \left(r_s(u,v) + \phi \sum_{s'} p_s^{uv}(s') \varphi_{s'}^{t_{s'v}}(j) \right) . \tag{3.24}$$

So, to guarantee that $\dfrac{\partial L(\lambda)}{\partial \varphi_s^{ut}(j)} = 0 \; \forall \lambda, j$ (i.e., minimizing (3.21) with respect to $\{\varphi_s^{ut}(j)\}_j$), the values of $\{\varphi_s^{ut}(j)\}_j$ can be set as (from (3.23) and (3.24))

$$\varphi_s^{ut}(j) \;=\; \sum_v p_s^v(\lambda^j) \left(r_s(u,v) + \phi \sum_{s'} p_s^{uv}(s') \varphi_{s'}^{t_{s'v}}(j) \right) . \tag{3.25}$$

Surprisingly, this equation specifies exactly the α-vector back-up operation for the discrete version of (3.5):

$$V_s(\hat{b}) \;=\; \max_u \sum_v \left\langle p_s^v, \hat{b} \right\rangle \left(r_s(u,v) + \phi \sum_{s'} p_s^{uv}(s') V_{s'}(\hat{b}_s^v) \right) , \tag{3.26}$$

where $\hat{b}$ is the discrete distribution over the set of samples $\{\lambda^j\}_j$ (i.e., $\sum_j \hat{b}(\lambda^j) = 1$). This implies that by choosing $\{\Psi_i(\lambda)\}_i$ as our basis functions, finding the corresponding "projected" solution for (3.5) is identical to solving (3.26), which can be easily implemented using the existing discrete POMDP solvers (e.g., [Pineau *et al.*, 2003]). This interestingly aligns with the idea of MC-BRL [Wang *et al.*, 2012] where a finite set of model candidates is sampled in advance to approximate the continuous model spectrum and consequently, cast BRL as a discrete POMDP problem.

3.3 Experiments and Discussion

This section evaluates the I-BRL framework using a set of benchmark problems. In particular, I-BRL is first evaluated in two small yet typical application domains which are frequently used in many existing single- and multi-agent RL works (Sections 3.3.1). Then, it is further tested in a more realistic domain modeled after a practical traffic problem (Section 3.3.2).

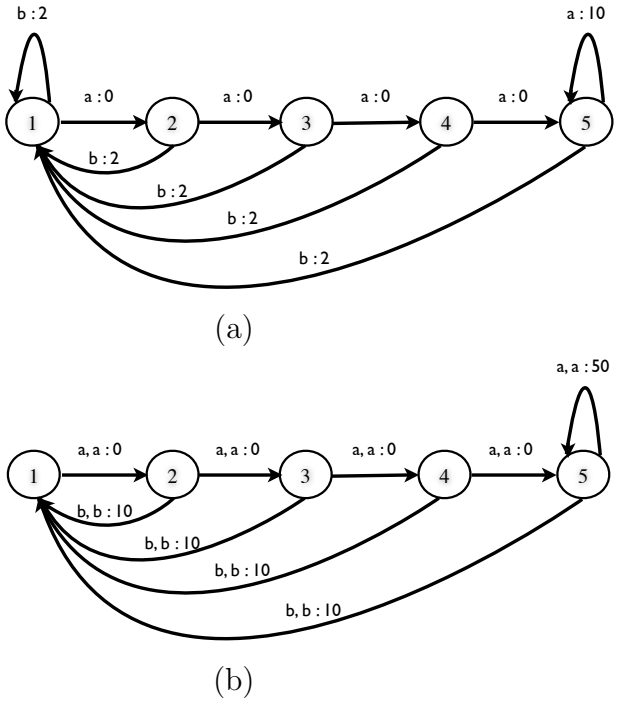


Figure 3.1: Chain problems: (a) Single-agent and (b) multi-agent versions.

3.3.1 Chain-World Problems

In this experiment, we consider both the single- [Dearden *et al.*, 1998; Poupart *et al.*, 2006; Wang *et al.*, 2012] and multi-agent [Chalkiadakis and Boutilier, 2003] versions of the Chain problem as depicted in Fig. 3.1 below. Specifically, the system consists

of a chain of 5 states and 2 possible actions $\{a, b\}$ which cause the agent's forward and backward transitions between states, respectively. In the single-agent version, the agent may slip with probability 0.2 while choosing an action and consequently incurs the effect of the other action (Fig. 3.1a). We experiment with both its Tied and Full parametric settings [Poupart *et al.*, 2006]: In the former setting, the agent is fully aware of the chain's transition structure except its slipping probability while in the latter setting, the chain's transition structure is completely unspecified.

Table 3.1: Average total (undiscounted) rewards of I-BRL, MC-BRL, BEETLE and BPVI ($\phi = 0.99$) for the single-agent Chain problem (Full and Tied versions) over 20 simulations, each of which lasts 100 steps.

Full Version	Total Reward	Simulation (sec)	Offline Planning (sec)
BEETLE	234.20 ± 14.75	26.95	220.88
MC-BRL (n = 100)	236.30 ± 18.81	160.16	300.00 [7]
I-BRL (n = 100)	244.40 ± 21.59	8.35	147.77
BPVI	282.80 ± 19.42	228.3	–
Tied Version	**Total Reward**	**Simulation (sec)**	**Offline Planning (sec)**
BEETLE	371.50 ± 18.03	3.46	30.24
MC-BRL (n = 300)	371.50 ± 18.03	434.21	300.00 [7]
I-BRL (n = 300)	360.60 ± 17.72	40.23	614.93
BPVI	161.90 ± 2.22	181.69	–

Table 3.2: Average total (discounted) rewards of I-BRL, MC-BRL, BEETLE and BPVI for the multi-agent Chain problem ($\phi = 0.85$) over 20 simulations, each of which lasts 100 steps and is averaged over 10 random opponents.

	Total Reward	Simulation (sec)	Offline Planning (sec)
BEETLE	8.19 ± 1.10	450.42	424.99
MC-BRL (n = 100)	5.87 ± 1.57	2261.40	300.00 [7]
I-BRL (n = 100)	8.19 ± 1.10	78.55	180.39
BPVI	5.32 ± 1.36	3642.00	–

In the multi-agent Chain problem, the agent can only move one step forward or go back to the initial state depending on whether it can coordinate with its opponent

on actions a or b at each stage of interaction. If both agents fail to coordinate on the same action, they remain at the current state and will not be rewarded. Otherwise, they will receive an immediate reward of 50 for coordinating on a in the last state and 10 for coordinating on b in any state except the first one (Fig. 3.1b). After each step, their payoffs are discounted by a constant factor of $0 < \phi < 1$.

For evaluation, we compare the performance of I-BRL with the state-of-the-art frameworks in both single- and (self-interested) multi-agent RL which include BPVI [Chalkiadakis and Boutilier, 2003], BEETLE [Poupart *et al.*, 2006], and MC-BRL [Wang *et al.*, 2012]. In particular, we report the average collected reward R as well as the online simulation and offline planning time of each framework when tested on the single- (Table 3.1) and multi-agent (Table 3.2) Chain problems for comparison. To achieve this, we run the offline phase of each algorithm, obtain a policy and simulate it in an online fashion to measure its performance: I-BRL and BEETLE plan their corresponding policies for 100 steps ahead while MC-BRL's anytime offline planner[7] is run up to 300 seconds; BPVI computes its policy during runtime and hence, does not require offline processing. Then, for the single-agent Chain problem, we evaluate each algorithm using 20 simulations with $h = 100$ steps in each simulation. For the multi-agent Chain problem, we additionally report the average performance of these works when tested against 10 different opponents whose behaviors are modeled as a set of probabilities $\theta_s = \{\theta_s^v\}_v$ (i.e., of selecting action $v \in \{a, b\}$ in state s). These opponents are independently and randomly generated from these Dirichlet distributions with the parameters $(\theta_s^a, \theta_s^b) \sim \text{Dir}(8, 2)$. Against each such opponent, we run 20 simulations ($h = 100$ steps each) to evaluate the average (discounted) total reward R collected by each framework.

[7] In this paper, the MC-BRL policy is computed offline by running the anytime POMDP solver of Kurniawati *et al.* [2008] (SARSOP) up to 300 seconds.

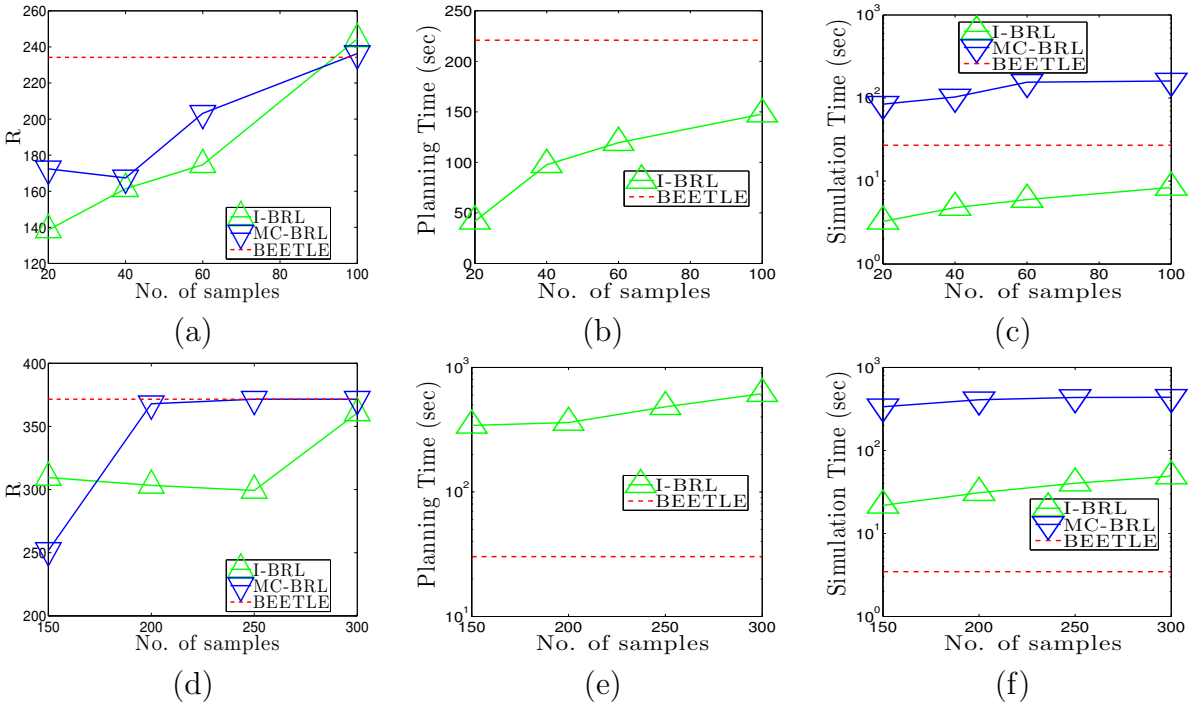


Figure 3.2: Graphs of the average performance, offline planning time and the total online simulation time (for all simulations) of I-BRL, MC-BRL and BEETLE (vs. the number of samples drawn during the offline planning phase) in the Full (a-c) and Tied (d-f) settings of the single-agent Chain problem.

From these results, it can be observed that I-BRL achieves good performance relative to the existing state-of-the-art algorithms in both settings (i.e., Full and Tied) of the single-agent Chain problem. Specifically, in the Full setting, I-BRL ($n = 100$) performs slightly better than both MC-BRL ($n = 100$) [Wang *et al.*, 2012] and BEETLE [Poupart *et al.*, 2006] while incurring significantly less expensive online processing cost (Table 3.1). Notably, I-BRL also uses less planning time than BEETLE in the Full setting of the single-agent Chain problem where the transition structure is completely unspecified which results in a large number of unknown parameters (i.e., larger belief dimension). This is expected since BEETLE's unit processing cost generally

depends on the complexity of analytically integrating a basis function with a sampled belief [Poupart *et al.*, 2006] which increases radically when the number of parameters increases. In contrast, I-BRL's processing cost only increases mildly in the number of parameters since its approximated integration (see Eqs. (3.15) and (3.16)) just involves the computation of $\{\Phi_s^i(\lambda^j)\}$ (Eq. 3.15) which can be cached once in advance for all future uses. Its processing cost, in fact, strongly depends on the number of samples drawn from the initial prior. Figs. 3.2b and 3.2c show that when we reduce the number of samples, I-BRL's offline and online processing cost drop at the cost of its performance degradation (Fig. 3.2a). Furthermore, Fig. 3.2a also shows that I-BRL's performance degrades faster than MC-BRL's which implies MC-BRL is more robust in terms of the performance quality when there are less samples. MC-BRL's online processing cost is, however, more expensive than both I-BRL and BEETLE (Fig. 3.2c). We suspect this is due to the difference between I-BRL's and MC-BRL's uses of the samples. While MC-BRL exclusively distributes its confidence probability among the finite number of sampled candidates and therefore, is less affected by the lack of samples as long as the true model is in the "proximity" of its samples, I-BRL's approximated integration in Eqs. (3.15) and (3.16), however, degrades significantly when there are not enough samples.

On the other hand, I-BRL's performance slightly loses to those of MC-BRL and BEETLE in the Tied version of the single-agent Chain problem in terms of the total collected reward but its online processing cost is, as expected, significantly less expensive than both those of MC-BRL and BPVI (Table 3.1). Fig. 3.2d also agrees with our previous observations that I-BRL and MC-BRL performance gradually approach that of BEETLE when we increase the number of samples at the cost of more intensive processing time (Figs. 3.2e and 3.2f). On a separate note, we like to point out that the reported online processing time of the tested algorithms is accumulated over 20 simu-

lations each of which lasts 100 steps. This amounts to 2000 online execution steps so if we divide the reported time by this number, the online processing time per step of the tested algorithms is actually negligible. In addition, we also evaluate I-BRL on the more commonly used experiment settings of the Chain problem [Poupart *et al.*, 2006; Wang *et al.*, 2012] to provide statistics easily comparable to the existing RL algorithms which are not covered in this thesis[8]. The corresponding results are reported in Table 3.3 below, which is consistent with our previous observations.

Table 3.3: Average total (undiscounted) rewards of I-BRL, MC-BRL, BEETLE and BPVI ($\phi = 0.99$) for the single-agent Chain problem (Full, Semi-Tied and Tied versions) over 500 simulations, each of which lasts 1000 steps.

	Full Version			Semi-Tied Version			Tied Version		
BEETLE	1754.00	±	42.00	3648.00	±	41.00	3650.00	±	41.00
I-BRL	1928.40	±	09.31	3030.72	±	11.34	3665.94	±	12.44
MC-BRL	1630.00	±	25.00	3603.00	±	32.00	3672.65	±	12.44
BPVI	3530.00	±	13.36	3302.32	±	11.93	2953.12	±	10.81

Finally, in the multi-agent Chain problem which adopts a more adverse rewarding scheme (see detail below), it is observed that I-BRL and BEETLE outperforms both BPVI and MC-BRL in terms of the total discounted reward (Table 3.2). This is expected because BPVI and MC-BRL, as mentioned in Section 2.1, appear to underestimate the risk of moving forward and forfeiting the opportunity to go backward to get more information and earn the small reward. Therefore, the chance of getting big reward (before it is severely discounted) is accidentally over-estimated due to BPVI's sub-optimal myopic information-gain function [Dearden *et al.*, 1998] and MC-BRL's exclusive confidence distribution on the sampled model candidates which hurts its performance if the majority of its sampled models are highly dissimilar to the true

[8]Our focus in this work is, however, not to compare I-BRL with the existing state-of-the-art RL algorithms on the Chain problem. Instead, we aim to highlight its efficiency in practical domains [Wang *et al.*, 2012] which do not admit FDM parameterization.

transition model, as "adversely" constructed for this experiment[9]. Consequently, this makes the expected gain of moving forward insufficient to compensate for the risk of doing so. In terms of the processing cost, it is again noticeable that I-BRL uses significantly less planning time than BEETLE and makes online decision faster than both BEETLE and MC-BRL (Table 3.2). This confirms and reinforces our previous conclusions regarding the performance of BEETLE, MC-BRL and I-BRL.

3.3.2 Intersection Navigation

In this section, we experiment on a realistic RL task of intersection navigation which is inspired from a near-miss accident during the 2007 DARPA Urban Challenge and modeled as a stochastic game[10] [Wang *et al.*, 2012]. For a brief description, let us consider the traffic situation illustrated in Fig. 3.3 below where two autonomous vehicles (marked A and B) are about to enter an intersection (I). The road segments are discretized into a uniform grid with cell size 5 m $\times$ 5 m and the speed of each vehicle is also discretized uniformly into 5 levels ranging from 0 m/s to 4 m/s. So, in each stage, the system's state can be characterized as a tuple $\{P_{\mathrm{A}}, P_{\mathrm{B}}, S_{\mathrm{A}}, S_{\mathrm{B}}\}$ specifying the current positions (P) and velocities (S) of A and B, respectively. In addition, our vehicle (A) can either accelerate ($+1$ m/s^2), decelerate (-1 m/s^2), or maintain its speed ($+0$ m/s^2) in each step while the other vehicle (B) changes its

[9] In this multi-agent Chain experiment, both I-BRL and MC-BRL draw the samples $\{(\theta_s^a, \theta_s^b)\}_s$ from $\mathrm{Dir}(1,1)$ while the simulated opponents are instead drawn from $\mathrm{Dir}(8,2)$ which statistically suggests the "backward" strategy. MC-BRL's exclusive distribution of confidence on its sampled models thus appears less cautious as compared to I-BRL in this adverse scenario.

[10] The RL formulation used in this section is adapted (by scaling down the problem size) from the original RL problem described in [Wang *et al.*, 2012].

speed based on the reactive model of Gipps [1981]:

$$\begin{aligned} v_{\text{safe}} &= S_{\text{B}} + \frac{\text{Distance}(P_{\text{A}}, P_{\text{B}}) - \tau S_{\text{B}}}{S_{\text{B}}/d + \tau} \\ v_{\text{des}} &= \min(4, S_{\text{B}} + a, v_{\text{safe}}) \\ S'_{\text{B}} &\sim \text{Uniform}(\max(0, v_{\text{des}} - \sigma a), v_{\text{des}}) \ . \end{aligned}$$

In this model, the driver's acceleration $a \in [0.5 \text{ m/s}^2, 3 \text{ m/s}^2]$, deceleration $d \in [-3 \text{ m/s}^2, -0.5 \text{ m/s}^2]$, reaction time $\tau \in [0.5\text{s}, 2\text{s}]$, and imperfection $\sigma \in [0, 1]$ are the unknown parameters distributed uniformly within the corresponding ranges. This parameterization can cover a variety of drivers' typical behaviors, as shown in a preliminary study. For a further understanding of these parameters, the readers are referred to [Gipps, 1981]. Besides, in each time step, each vehicle $\text{X} \in \{\text{A}, \text{B}\}$ moves from its current cell P_{X} to the next cell P'_{X} with probability $1/t$ and remains in the same cell with probability $1 - 1/t$ where t is the expected time to move forward one cell from the current position with respect to the current speed (e.g., $t = 5/S_{\text{X}}$). Thus, in general, the underlying stochastic game has $6 \times 6 \times 5 \times 5 = 900$ states (i.e., each vehicle has 6 possible positions and 5 levels of speed), which is significantly larger than the settings in previous experiments. In each state, our vehicle has 3 actions, as mentioned previously, while the other vehicle has 5 actions corresponding to 5 levels of speed according to the reactive model.

The goal for our vehicle in this domain is to learn the other vehicle's reactive model and adjust its navigation strategy accordingly such that there is no collision and the time spent to cross the intersection is minimized. To achieve this goal, we penalize our vehicle in each step by -1 and reward it with 50 when it successfully crosses the intersection. If it collides with the other vehicle (at I), we penalize it by -250. The discount factor is set as 0.99. We evaluate the performance of I-BRL ($n = 100$) in this

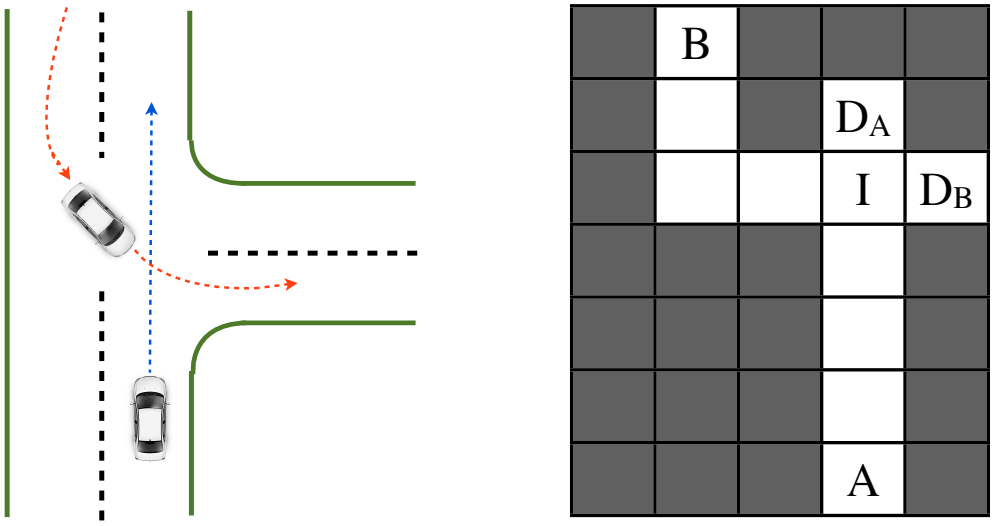


Figure 3.3: (Left) A near-miss accident during the 2007 DARPA Urban Challenge, and (Right) the discretized environment: A and B move towards destinations D_A and D_B while avoiding collision at I. Shaded areas are not passable.

problem against 100 different sets of reactive parameters (for the other vehicle) generated uniformly from the above ranges. Its policy is obtained by planning offline for 10 steps ahead which incurs 87 minutes. Against each set of parameters, we run 20 simulations ($h = 100$ steps each) to estimate our vehicle's average performance[11] R. In particular, we compare our algorithm's average performance (over a total number of 2000 simulations) against those of an omniscience vehicle (**UPPER-BOUND**) who knows exactly the reactive parameters before each simulation, and two other vehicles employing MC-BRL ($n = 100$), which plans offline for 1.5 hours [Wang *et al.*, 2012] (**MC-BRL**), and BPVI [Chalkiadakis and Boutilier, 2003] (**BPVI**), respectively.

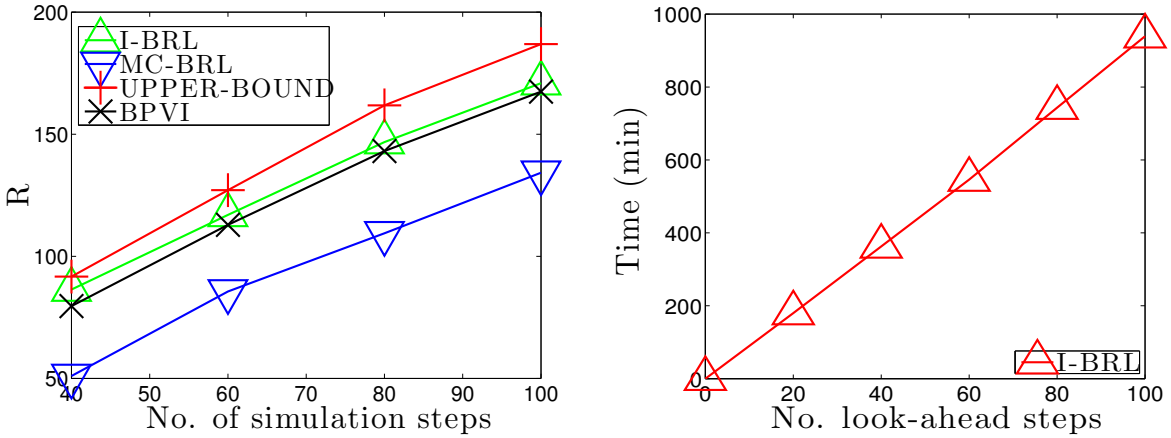
The results are shown in Fig. 3.4a and Table 3.4: It can be observed that our vehicle's average performance (over 2000 simulations) is better than both those of the MC-

[11] After our vehicle successfully crosses the intersection, the system's state is reset to the default state in Fig. 3.3 (Right).

	Reward	Travel Time	Accident (%)	Intersections
UPPER-BOUND	186.88 ± 3.75	09.66 (steps)	2.46	17783
MC-BRL (n = 100)	134.22 ± 4.38	10.32 (steps)	4.53	16660
I-BRL (n = 100)	170.88 ± 3.24	10.10 (steps)	2.80	16924
BPVI	167.43 ± 3.40	10.22 (steps)	2.58	16785

Table 3.4: The number of cleared intersections (in 2000 simulations), accident rates and average traveling time to navigate through 1 intersection as well as the total discounted rewards of the I-BRL, BPVI, MC-BRL and omniscience vehicles.

BRL- and BPVI-based vehicles. In particular, our vehicle manages to significantly reduce the performance gap (in terms of the total rewards) between the omniscience and the MC-BRL vehicles by more than half (Fig. 3.4). Remarkably, I-BRL manages to safely clear more intersections than both MC-BRL and BPVI; its accident rate is significantly smaller than MC-BRL's and comparable to both the omniscience and BPVI vehicles. In addition, I-BRL also uses less time (on average) than MC-BRL and BPVI to navigate through an intersection (Table 3.4).



(a) (b)

Figure 3.4: (a) Performance comparison between our vehicle (I-BRL), the omniscience (UPPER) vehicle and two other vehicles which employ BPVI and MC-BRL, respectively ($\phi = 0.99$); (b) I-BRL's offline planning time up to 100 steps ahead.

In fact, the difference in performance between our vehicle and the omniscience vehicle is expected as the omniscience vehicle always takes the optimal step from the beginning (since it knows the reactive parameters in advance) while our vehicle has to take cautious steps (by maintaining a slow speed) before it feels confident with the information collected during interaction. On the other hand, MC-BRL appears to be misled by its exclusive confidence on its a priori sampled models which unfortunately results in a much less competitive performance: Both its average traveling time (per intersection) and its accident rate are highest among the tested vehicles (Table 3.4). Besides, since the uniform prior over the reactive parameters $\lambda = \{a, d, \tau, \sigma\}$ is not a conjugate prior for the other vehicle's behavior model $\theta_s(v) = p(v|s, \lambda)$, the BPVI-based vehicle has to directly maintain and update its belief using FDM: $\lambda = \{\theta_s\}_s$ with $\theta_s = \{\theta_s^v\}_v \sim \mathrm{Dir}(\{n_s^v\}_v)$ (Section 3.1), instead of $\lambda = \{a, d, \tau, \sigma\}$. However, FDM implicitly assumes that $\{\theta_s\}_s$ are statistically independent, which is not true in this case since all θ_s are actually related by $\{a, d, \tau, \sigma\}$. Unfortunately, BPVI cannot exploit this information to generalize the other vehicle's behavior across different states due to its restrictive FDM (i.e., independent multinomial likelihoods with separate Dirichlet priors), thus performing marginally worse than I-BRL in terms of the expected total reward.

Chapter 4

Nonmyopic ϵ-Bayes-Optimal Active Learning (ϵ-BAL)

This chapter presents a novel nonmyopic *ϵ-Bayes-optimal* (ϵ-BAL) approach to optimize the fundamental exploration-exploitation trade-off in active learning of Gaussian processes (Section 4.2.1). Unlike the existing works in the literature which have primarily developed myopic/greedy algorithms [Diggle, 2006; Houlsby *et al.*, 2012; Park and Pillow, 2012; Zimmerman, 2006; Ouyang *et al.*, 2014] or performed exploration and exploitation separately [Krause and Guestrin, 2007], ϵ-BAL preserves and exploits the principled Bayesian sequential decision framework to jointly optimize the trade-off and consequently does not incur their limitations (Section 2.2). In particular, although the exact Bayes-optimal policy to the active sensing problem cannot be derived [Solomon and Zacks, 1970], we show that it is in fact possible to solve for a nonmyopic *ϵ-Bayes-optimal active learning* (ϵ-BAL) policy (Sections 4.2.2 and 4.2.3) given a user-defined bound ϵ, which is the main contribution of our work here. In other words, our proposed ϵ-BAL policy can approximate the optimal expected active sensing performance arbitrarily closely (i.e., within an arbitrary loss bound ϵ).

In contrast, the algorithm of Krause and Guestrin [2007] can only yield a sub-optimal performance bound[1]. To meet the real-time requirement in time-critical applications, we then propose an asymptotically ϵ-optimal, branch-and-bound anytime algorithm based on ϵ-BAL with performance guarantee (Section 4.2.4). We empirically demonstrate using both synthetic and real-world datasets that, with limited budget, our proposed approach outperforms state-of-the-art algorithms (Section 4.3).

4.1 Modeling Spatial Phenomena with Gaussian Processes (GPs)

The GP can be used to model a spatial phenomenon of interest as follows: The phenomenon is defined to vary as a realization of a GP. Let $\mathcal{X}$ denote a set of sampling locations representing the domain of the phenomenon such that each location $x \in \mathcal{X}$ is associated with a realized (random) measurement z_x (Z_x) if x is observed/sampled (unobserved). Let $Z_{\mathcal{X}} \triangleq \{Z_x\}_{x\in\mathcal{X}}$ denote a GP, that is, every finite subset of $Z_{\mathcal{X}}$ has a multivariate Gaussian distribution [Chen *et al.*, 2013b; Rasmussen and Williams, 2006]. The GP is fully specified by its *prior* mean $\mu_x \triangleq \mathbb{E}[Z_x]$ and covariance $\sigma_{xx'|\lambda} \triangleq \mathrm{cov}[Z_x, Z_{x'}|\lambda]$ for all $x, x' \in \mathcal{X}$, the latter of which characterizes the spatial correlation structure of the phenomenon and can be defined using a covariance function parameterized by λ. A common choice is the squared exponential covariance function:

$$\sigma_{xx'|\lambda} \triangleq (\sigma_s^{\lambda})^2 \exp\left(-\frac{1}{2}\sum_{i=1}^{P}\left(\frac{[s_x]_i - [s_{x'}]_i}{\ell_i^{\lambda}}\right)^2\right) + (\sigma_n^{\lambda})^2\delta_{xx'} \quad (4.1)$$

[1]Its induced policy is guaranteed not to achieve worse than the optimal performance by more than a factor of $1/e$.

where $[s_x]_i$ ($[s_{x'}]_i$) is the i-th component of the P-dimensional feature vector s_x ($s_{x'}$), the set of realized parameters $\lambda \triangleq \left\{\sigma_n^\lambda, \sigma_s^\lambda, \ell_1^\lambda, \ldots, \ell_P^\lambda\right\} \in \Lambda$ are, respectively, the square root of noise variance, square root of signal variance, and length-scales, and $\delta_{xx'}$ is a Kronecker delta that is 1 if $x = x'$ and 0 otherwise. Supposing λ is known and a set $z_\mathcal{D}$ of realized measurements is available for some set $\mathcal{D} \subset \mathcal{X}$ of observed locations, the GP can exploit these observations to predict the measurement for any unobserved location $x \in \mathcal{X}\backslash\mathcal{D}$ as well as provide its corresponding predictive uncertainty using the Gaussian predictive distribution $p(z_x|z_\mathcal{D}, \lambda) \sim \mathcal{N}(\mu_{x|\mathcal{D},\lambda}, \sigma_{xx|\mathcal{D},\lambda})$ with the following *posterior* mean and variance, respectively:

$$\mu_{x|\mathcal{D},\lambda} \triangleq \mu_x + \Sigma_{x\mathcal{D}|\lambda}\Sigma_{\mathcal{D}\mathcal{D}|\lambda}^{-1}(z_\mathcal{D} - \mu_\mathcal{D}) \tag{4.2}$$

$$\sigma_{xx|\mathcal{D},\lambda} \triangleq \sigma_{xx|\lambda} - \Sigma_{x\mathcal{D}|\lambda}\Sigma_{\mathcal{D}\mathcal{D}|\lambda}^{-1}\Sigma_{\mathcal{D}x|\lambda} \tag{4.3}$$

where, with a slight abuse of notation, $z_\mathcal{D}$ is to be perceived as a column vector in (4.2), $\mu_\mathcal{D}$ is a column vector with mean components $\mu_{x'}$ for all $x' \in \mathcal{D}$, $\Sigma_{x\mathcal{D}|\lambda}$ is a row vector with covariance components $\sigma_{xx'|\lambda}$ for all $x' \in \mathcal{D}$, $\Sigma_{\mathcal{D}x|\lambda}$ is the transpose of $\Sigma_{x\mathcal{D}|\lambda}$, and $\Sigma_{\mathcal{D}\mathcal{D}|\lambda}$ is a covariance matrix with components $\sigma_{ux'|\lambda}$ for all $u, x' \in \mathcal{D}$. When the spatial correlation structure (i.e., λ) is not known, a probabilistic belief $b_\mathcal{D}(\lambda) \triangleq p(\lambda|z_\mathcal{D})$ can be maintained/tracked over all possible λ and updated using Bayes' rule to the posterior belief $b_{\mathcal{D}\cup\{x\}}(\lambda)$ given a newly available measurement z_x:

$$b_{\mathcal{D}\cup\{x\}}(\lambda) \propto p(z_x|z_\mathcal{D}, \lambda)\ b_\mathcal{D}(\lambda)\ . \tag{4.4}$$

Using belief $b_\mathcal{D}$, the predictive distribution $p(z_x|z_\mathcal{D})$ can be obtained by marginalizing out the unknown parameters λ:

$$p(z_x|z_\mathcal{D}) = \sum_{\lambda\in\Lambda} p(z_x|z_\mathcal{D}, \lambda)\ b_\mathcal{D}(\lambda)\ . \tag{4.5}$$

4.2 Nonmyopic ϵ-Bayes-Optimal Active Learning

4.2.1 Problem Formulation

To cast active sensing as a Bayesian sequential decision problem, let us first define a sequential active sensing/learning policy π given a budget of N sampling locations: Specifically, the policy $\pi \triangleq \{\pi_n\}_{n=1}^N$ is structured to sequentially decide the next location $\pi_n(z_\mathcal{D}) \in \mathcal{X} \setminus \mathcal{D}$ to be observed at each stage n based on the current observations $z_\mathcal{D}$ over a finite planning horizon of N stages. Recall from Section 2.2 that the active sensing problem involves planning/deciding the most informative locations to be observed for minimizing the predictive uncertainty of the unobserved areas of a phenomenon. To achieve this, we use the entropy criterion [Cover and Thomas, 1991] to measure the informativeness and predictive uncertainty. Then, the value under a policy π is defined to be the joint entropy of its selected observations when starting with some prior observations $z_{\mathcal{D}_0}$ and following π thereafter:

$$V_1^\pi(z_{\mathcal{D}_0}) \triangleq \mathbb{H}[Z_\pi|z_{\mathcal{D}_0}] \triangleq -\int p(z_\pi|z_{\mathcal{D}_0}) \log p(z_\pi|z_{\mathcal{D}_0})\, \mathrm{d}z_\pi \tag{4.6}$$

where Z_π (z_π) is the set of random (realized) measurements taken by policy π and $p(z_\pi|z_{\mathcal{D}_0})$ is defined in a similar manner to (4.5). To solve the active sensing problem, the notion of Bayes-optimality[2] is exploited for selecting observations of largest possible joint entropy with respect to all possible induced sequences of future beliefs (starting from initial prior belief $b_{\mathcal{D}_0}$) over candidate sets of model parameters λ, as detailed next. Formally, this entails choosing a sequential policy π to maximize $V_1^\pi(z_{\mathcal{D}_0})$ (4.6), which we call the *Bayes-optimal active learning* (BAL) policy π^*. That is, $V_1^*(z_{\mathcal{D}_0}) \triangleq V_1^{\pi^*}(z_{\mathcal{D}_0}) = \max_\pi V_1^\pi(z_{\mathcal{D}_0})$. When π^* is plugged into (4.6),

[2]Bayes-optimality is previously studied in reinforcement learning whose developed theories [Poupart *et al.*, 2006; Hoang and Low, 2013a] cannot be applied here because their assumptions of discrete-valued observations and Markov property do not hold.

the following N-stage Bellman equations result from the chain rule for entropy:

$$\begin{aligned} Q_n^*(z_\mathcal{D}, x) &\triangleq \mathbb{H}\left[Z_x|z_\mathcal{D}\right] + \mathbb{E}\left[V_{n+1}^*\left(z_\mathcal{D} \cup \{Z_x\}\right)|z_\mathcal{D}\right] \\ V_n^*(z_\mathcal{D}) &\triangleq \max_{x \in \mathcal{X}\setminus\mathcal{D}} Q_n^*(z_\mathcal{D}, x) \end{aligned} \tag{4.7}$$

with $\mathbb{H}\left[Z_x|z_\mathcal{D}\right] \triangleq -\int p(z_x|z_\mathcal{D}) \log p(z_x|z_\mathcal{D}) \mathrm{d}z_x$ for stage $n = 1, \ldots, N$ where $p(z_x|z_\mathcal{D})$ is defined in (4.5) and the expectation terms are omitted from the right-hand side (RHS) expressions of V_N^* and Q_N^* at stage N. At each stage, the belief $b_\mathcal{D}(\lambda)$ is needed to compute $Q_n^*(z_\mathcal{D}, x)$ in (4.7) and can be uniquely determined from initial prior belief $b_{\mathcal{D}_0}$ and observations $z_{\mathcal{D}\setminus\mathcal{D}_0}$ using (4.4)[3].

To understand how π^* jointly and naturally optimizes the exploration-exploitation trade-off, its selected location $\pi_n^*(z_\mathcal{D}) = \arg\max_{x \in \mathcal{X}\setminus\mathcal{D}} Q_n^*(z_\mathcal{D}, x)$ at each stage n affects both the immediate payoff $\mathbb{H}\left[Z_{\pi_n^*(z_\mathcal{D})}|z_\mathcal{D}\right]$ given current belief $b_\mathcal{D}$ (i.e., exploitation) as well as the posterior belief $b_{\mathcal{D}\cup\{\pi_n^*(z_\mathcal{D})\}}$, the latter of which influences expected future payoff $\mathbb{E}[V_{n+1}^*(z_\mathcal{D} \cup \{Z_{\pi_n^*(z_\mathcal{D})}\})|z_\mathcal{D}]$ and builds in the information gathering option (i.e., exploration). Interestingly, the work of Low *et al.* [2009] has revealed that the above recursive formulation (4.7) can be perceived as the sequential variant of the well-known maximum entropy sampling problem [Shewry and Wynn, 1987] and established an equivalence result that the maximum-entropy observations selected by π^* achieve a dual objective of minimizing the posterior joint entropy (i.e., predictive uncertainty) remaining in the unobserved locations of the phenomenon. Unfortunately, the BAL policy π^* cannot be derived exactly because the stage-wise entropy and expectation terms in (4.7) cannot be evaluated in closed form [Huber *et al.*, 2008] due to an uncountable set of candidate observations and unknown model parameters

[3]For practical implementation, $b_\mathcal{D}(\lambda)$ can instead be updated incrementally at each stage and included as a component of the state to be passed on to the next stage.

λ (Section 2.2). To overcome this difficulty, we show in the next subsection how it is possible to solve for an ϵ-BAL policy π_ϵ, that is, the joint entropy of its selected observations closely approximates that of π^* within an arbitrary loss bound $\epsilon > 0$.

4.2.2 ϵ-BAL Policy

The key idea underlying the design and construction of our proposed nonmyopic ϵ-BAL policy π^ϵ is to approximate the entropy and expectation terms in (4.7) at every stage using a form of truncated sampling to be described next:

Definition 1 (τ-Truncated Observation)**.** *Define random measurement $\widehat{Z}_x$ by truncating Z_x at $-\widehat{\tau}$ and $\widehat{\tau}$ as follows:*

$$\widehat{Z}_x \triangleq \begin{cases} -\widehat{\tau} & \text{if } Z_x \leq -\widehat{\tau}, \\ Z_x & \text{if } -\widehat{\tau} < Z_x < \widehat{\tau}, \\ \widehat{\tau} & \text{if } Z_x \geq \widehat{\tau}. \end{cases}$$

Then, $\widehat{Z}_x$ has a distribution of mixed type [Soong, 2004] with its continuous component defined as $f(\widehat{Z}_x = z_x|z_\mathcal{D}) \triangleq p(Z_x = z_x|z_\mathcal{D})$ for $-\widehat{\tau} < z_x < \widehat{\tau}$ and its discrete component defined as $f(\widehat{Z}_x = \widehat{\tau}|z_\mathcal{D}) \triangleq P(Z_x \geq \widehat{\tau}|z_\mathcal{D}) = \int_{\widehat{\tau}}^{\infty} p(Z_x = z_x|z_\mathcal{D})\mathrm{d}z_x$ and $f(\widehat{Z}_x = -\widehat{\tau}|z_\mathcal{D}) \triangleq P(Z_x \leq -\widehat{\tau}|z_\mathcal{D}) = \int_{-\infty}^{-\widehat{\tau}} p(Z_x = z_x|z_\mathcal{D})\mathrm{d}z_x$. Let $\overline{\mu}(\mathcal{D}, \Lambda) \triangleq \max_{x \in \mathcal{X}\setminus\mathcal{D}, \lambda \in \Lambda} \mu_{x|\mathcal{D},\lambda}$, $\underline{\mu}(\mathcal{D}, \Lambda) \triangleq \min_{x \in \mathcal{X}\setminus\mathcal{D}, \lambda \in \Lambda} \mu_{x|\mathcal{D},\lambda}$, and

$$\widehat{\tau} \triangleq \max\left(\left|\underline{\mu}(\mathcal{D}, \Lambda) - \tau\right|, \left|\overline{\mu}(\mathcal{D}, \Lambda) + \tau\right|\right) \tag{4.8}$$

for some $0 \leq \tau \leq \widehat{\tau}$. Then, a realized measurement of $\widehat{Z}_x$ is said to be a τ-truncated observation for location x.

Specifically, given that a set $z_\mathcal{D}$ of realized measurements is available, a finite set of S τ-truncated observations $\{z_x^i\}_{i=1}^S$ can be generated for every candidate location

$x \in \mathcal{X} \setminus \mathcal{D}$ at each stage n by identically and independently sampling from $p(z_x|z_{\mathcal{D}})$ (4.5) and then truncating each of them according to $z_x^i \leftarrow z_x^i \min\left(|z_x^i|, \widehat{\tau}\right)/|z_x^i|$. These generated τ-truncated observations[4] can be exploited for approximating V_n^* (4.7) through the following Bellman equations:

$$\begin{aligned} V_n^\epsilon(z_{\mathcal{D}}) &\triangleq \max_{x \in \mathcal{X}\setminus\mathcal{D}} Q_n^\epsilon(z_{\mathcal{D}}, x) \\ Q_n^\epsilon(z_{\mathcal{D}}) &\triangleq \frac{1}{S}\sum_{i=1}^{S}\left(-\log p\left(z_x^i|z_{\mathcal{D}}\right) + V_{n+1}^\epsilon\left(z_{\mathcal{D}} \cup \left\{z_x^i\right\}\right)\right) \end{aligned} \tag{4.9}$$

for stage $n = 1, \ldots, N$ such that there is no V_{N+1}^ϵ term on the RHS expression of Q_N^ϵ at stage N. Like the BAL policy π^* (Section 4.2.1), the location $\pi_n^\epsilon(z_{\mathcal{D}}) = \arg\max_{x \in \mathcal{X}\setminus\mathcal{D}} Q_n^\epsilon(z_{\mathcal{D}}, x)$ selected by our ϵ-BAL policy π^ϵ at each stage n also jointly and naturally optimizes the trade-off between exploitation (i.e., by maximizing immediate payoff $S^{-1}\sum_{i=1}^{S} -\log p(z^i_{\pi_n^\epsilon(z_{\mathcal{D}})}|z_{\mathcal{D}})$ given the current belief $b_{\mathcal{D}}$) vs. exploration (i.e., by improving posterior belief $b_{\mathcal{D}\cup\{\pi_n^\epsilon(z_{\mathcal{D}})\}}$ to maximize average future payoff $S^{-1}\sum_{i=1}^{S} V_{n+1}^\epsilon(z_{\mathcal{D}} \cup \{z^i_{\pi_n^\epsilon(z_{\mathcal{D}})}\})$). Unlike the deterministic BAL policy π^*, our ϵ-BAL policy π^ϵ is stochastic due to its use of the above truncated sampling procedure.

4.2.3 Theoretical Analysis

The main difficulty in analyzing the active sensing performance of our stochastic ϵ-BAL policy π^ϵ (i.e., relative to that of BAL policy π^*) lies in determining how its ϵ-Bayes optimality can be guaranteed by choosing appropriate values of the truncated sampling parameters S and τ (Section 4.2.2). To achieve this, we have to formally understand how S and τ can be specified and varied in terms of the user-defined loss bound ϵ, budget of N sampling locations, domain size $|\mathcal{X}|$ of the phenomenon,

[4]The reason for using truncation may not be obvious to a reader at this time because it is motivated by a technical necessity for theoretically guaranteeing the performance of our ϵ-BAL policy π^ϵ (see Remark 1 after Lemma 2 in Section 4.2.3) rather than a conceptual intuition.

and properties/parameters characterizing the spatial correlation structure of the phenomenon (Section 4.1), as detailed below.

The first step is to show that Q_n^ϵ (4.9) is in fact a good approximation of Q_n^* (4.7) for some chosen values of S and τ. There are two sources of error arising in such an approximation: (a) In the truncated sampling procedure (Section 4.2.2), only a finite set of τ-truncated observations is generated for approximating the stage-wise entropy and expectation terms in (4.7), and (b) computing Q_n^ϵ does not involve utilizing the values of V_{n+1}^* but that of its approximation V_{n+1}^ϵ instead. To facilitate capturing the error due to finite truncated sampling described in (a), the following intermediate function is introduced:

$$W_n^*(z_\mathcal{D}, x) \triangleq \frac{1}{S}\sum_{i=1}^{S}\left(-\log p\left(z_x^i|z_\mathcal{D}\right) + V_{n+1}^*\left(z_\mathcal{D}\cup\{z_x^i\}\right)\right) \tag{4.10}$$

for stage $n = 1, \ldots, N$ such that there is no V_{N+1}^* term on the RHS expression of W_N^* at stage N. The first lemma below reveals that if the error $|Q_n^*(z_\mathcal{D}, x) - W_n^*(z_\mathcal{D}, x)|$ due to finite truncated sampling can be bounded for all tuples $(n, z_\mathcal{D}, x)$ generated at stage $n = n', \ldots, N$ by (4.9) to compute $V_{n'}^\epsilon$ for $1 \leq n' \leq N$, then $Q_{n'}^\epsilon$ (4.9) can approximate $Q_{n'}^*$ (4.7) arbitrarily closely:

Lemma 1. *Suppose that a set $z_{\mathcal{D}'}$ of observations, a budget of $N - n' + 1$ sampling locations for $1 \leq n' \leq N$, $S \in \mathbb{Z}^+$, and $\gamma > 0$ are given. If*

$$|Q_n^*(z_\mathcal{D}, x) - W_n^*(z_\mathcal{D}, x)| \leq \gamma \tag{4.11}$$

for all tuples $(n, z_\mathcal{D}, x)$ generated at stage $n = n', \ldots, N$ by (4.9) to compute $V_{n'}^\epsilon(z_{\mathcal{D}'})$,

then, for all $x' \in \mathcal{X} \setminus \mathcal{D}'$,

$$|Q^*_{n'}(z_{\mathcal{D}'}, x') - Q^\epsilon_{n'}(z_{\mathcal{D}'}, x')| \leq (N - n' + 1)\gamma \ . \tag{4.12}$$

Its proof is given in Appendix B.1. The next two lemmas show that, with high probability, the error $|Q^*_n(z_\mathcal{D}, x) - W^*_n(z_\mathcal{D}, x)|$ due to finite truncated sampling can indeed be bounded from above by γ (4.11) for all tuples $(n, z_\mathcal{D}, x)$ generated at stage $n = n', \ldots, N$ by (4.9) to compute $V^\epsilon_{n'}$ for $1 \leq n' \leq N$:

Lemma 2. *Suppose that a set* $z_{\mathcal{D}'}$ *of observations, a budget of* $N - n' + 1$ *sampling locations for* $1 \leq n' \leq N$, $S \in \mathbb{Z}^+$, *and* $\gamma > 0$ *are given. For all tuples* $(n, z_\mathcal{D}, x)$ *generated at stage* $n = n', \ldots, N$ *by* (4.9) *to compute* $V^\epsilon_{n'}(z_{\mathcal{D}'})$,

$$P\Big(\ |Q^*_n(z_\mathcal{D}, x) - W^*_n(z_\mathcal{D}, x)| \leq \gamma\Big) \ \geq \ 1 - 2\exp\left(-\frac{2S\gamma^2}{T^2}\right)$$

where $T \triangleq \mathcal{O}\left(\dfrac{N^2\kappa^{2N}\tau^2}{\sigma_n^2} + N\log\dfrac{\sigma_o}{\sigma_n} + \log|\Lambda|\right)$ *by setting*[5]

$$\tau \ = \ \mathcal{O}\left(\sigma_o\sqrt{\log\left(\frac{\sigma_o^2}{\gamma}\left(\frac{N^2\kappa^{2N} + \sigma_o^2}{\sigma_n^2} + N\log\frac{\sigma_o}{\sigma_n} + \log|\Lambda|\right)\right)}\right)$$

with κ, σ_n^2, *and* σ_o^2 *defined as follows:*

$$\kappa \ \triangleq \ 1 + 2\max_{x', u \in \mathcal{X}\setminus\mathcal{D}: x' \neq u, \lambda \in \Lambda, \mathcal{D}} \left|\sigma_{x'u|\mathcal{D},\lambda}\right| / \sigma_{uu|\mathcal{D},\lambda}, \tag{4.13}$$

$$\sigma_n^2 \ \triangleq \ \min_{\lambda\in\Lambda} (\sigma_n^\lambda)^2, \quad \textit{and} \quad \sigma_o^2 \triangleq \max_{\lambda\in\Lambda} (\sigma_s^\lambda)^2 + (\sigma_n^\lambda)^2. \tag{4.14}$$

Refer to Appendix B.2 for its proof.

[5]To simplify notations, the constants involved in computing the exact values of T, S, & τ are omitted; they are straightforward to obtain, albeit tedious, by following the derivation in our proofs.

Remark 1. Deriving such a probabilistic bound in Lemma 2 typically involves the use of concentration inequalities for the sum of independent *bounded* random variables like the Hoeffding's, Bennett's, or Bernstein's inequalities. However, since the originally Gaussian distributed observations are *unbounded*, sampling from $p(z_x|z_{\mathcal{D}})$ (4.5) without truncation will generate unbounded versions of $\{z_x^i\}_{i=1}^S$ and consequently make each summation term $-\log p(z_x^i|z_{\mathcal{D}}) + V_{n+1}^*(z_{\mathcal{D}} \cup \{z_x^i\})$ on the RHS expression of W_n^* (4.10) unbounded, hence invalidating the use of these concentration inequalities. To resolve this complication, our trick is to exploit the truncated sampling procedure (Section 4.2.2) to generate *bounded* τ-truncated observations (Definition 1) (i.e., $|z_x^i| \leq \widehat{\tau}$ for $i = 1, \ldots, S$), thus resulting in each summation term $-\log p(z_x^i|z_{\mathcal{D}}) + V_{n+1}^*(z_{\mathcal{D}} \cup \{z_x^i\})$ being bounded (Appendix B.2). This enables our use of Hoeffding's inequality to derive the probabilistic bound.

Remark 2. It can be observed from Lemma 2 that the amount of truncation has to be reduced (i.e., higher chosen value of τ) when (a) a tighter bound γ on the error $|Q_n^*(z_{\mathcal{D}}, x) - W_n^*(z_{\mathcal{D}}, x)|$ due to finite truncated sampling is desired, (b) a greater budget of N sampling locations is available, (c) a larger space Λ of candidate model parameters is preferred, (d) the spatial phenomenon varies with more intensity and less noise (i.e., assuming all candidate signal and noise variance parameters, respectively, $(\sigma_s^\lambda)^2$ and $(\sigma_n^\lambda)^2$ are specified close to the true large signal and small noise variances), and (e) its spatial correlation structure yields a bigger κ.

To elaborate on (e), note that Lemma 2 still holds for any value of κ larger than that set in (4.13): Since $|\sigma_{x'u|\mathcal{D},\lambda}|^2 \leq \sigma_{x'x'|\mathcal{D},\lambda}\sigma_{uu|\mathcal{D},\lambda}$ for all $x' \neq u \in \mathcal{X} \setminus \mathcal{D}$ due to the symmetric positive-definiteness of $\Sigma_{(\mathcal{X}\setminus\mathcal{D})(\mathcal{X}\setminus\mathcal{D})|\mathcal{D},\lambda}$ [Rue and Held, 2005], we can set κ

as following:

$$\kappa \;=\; 1 + 2 \max_{x', u \in \mathcal{X} \setminus \mathcal{D}, \lambda \in \Lambda, \mathcal{D}} \sqrt{\sigma_{x'x'|\mathcal{D},\lambda} / \sigma_{uu|\mathcal{D},\lambda}}$$

Then, supposing all candidate length-scales are specified close to the true length-scales, a phenomenon with extreme length-scales tending to 0 (i.e., with white-noise process measurements) or ∞ (i.e., with constant measurements) will produce highly similar $\sigma_{x'x'|\mathcal{D},\lambda}$ for all $x' \in \mathcal{X} \setminus \mathcal{D}$, thus resulting in smaller κ and hence smaller τ.

Remark 3. Alternatively, it can be proven that Lemma 2 and the subsequent results hold by setting $\kappa = 1$ if a certain structural property of the spatial correlation structure (i.e., for all $z_{\mathcal{D}}$ ($\mathcal{D} \subseteq \mathcal{X}$) and $\lambda \in \Lambda$, $\Sigma_{\mathcal{D}\mathcal{D}|\lambda}$ is diagonally dominant) is satisfied, as shown in Lemma 10 (Appendix B.3). Consequently, the κ term can be removed from T and τ.

Lemma 3. *Suppose that a set $z_{\mathcal{D}'}$ of observations, a budget of $N - n' + 1$ sampling locations for $1 \leq n' \leq N$, $S \in \mathbb{Z}^+$, and $\gamma > 0$ are given. The probability that $|Q^*_n(z_{\mathcal{D}}, x) - W^*_n(z_{\mathcal{D}}, x)| \leq \gamma$ (4.11) holds for all tuples $(n, z_{\mathcal{D}}, x)$ generated at stage $n = n', \ldots, N$ by (4.9) to compute $V^{\epsilon}_{n'}(z_{\mathcal{D}'})$ is at least $1 - 2(S|\mathcal{X}|)^N \exp(-2S\gamma^2/T^2)$ where T is previously defined in Lemma 2.*

Its proof is found in Appendix B.3.

The first step is concluded with our first main result, which follows from Lemmas 1 and 3. Specifically, it chooses the values of S and τ such that the probability of Q^{ϵ}_n (4.9) approximating Q^*_n (4.7) poorly (i.e., $|Q^*_n(z_{\mathcal{D}}, x) - Q^{\epsilon}_n(z_{\mathcal{D}}, x)| > N\gamma$) can be bounded from above by a given $0 < \delta < 1$:

Theorem 5. *Suppose that a set $z_{\mathcal{D}}$ of observations, a budget of $N - n + 1$ sampling*

locations for $1 \leq n \leq N$, $\gamma > 0$, *and* $0 < \delta < 1$ *are given. The probability that* $|Q_n^*(z_\mathcal{D}, x) - Q_n^\epsilon(z_\mathcal{D}, x)| \leq N\gamma$ *holds for all* $x \in \mathcal{X} \setminus \mathcal{D}$ *is at least* $1 - \delta$ *by setting*

$$S = \mathcal{O}\left(\frac{T^2}{\gamma^2}\left(N \log \frac{N|\mathcal{X}|T^2}{\gamma^2} + \log \frac{1}{\delta}\right)\right)$$

where T *is previously defined in Lemma 2. By assuming* N, $|\Lambda|$, σ_o, σ_n, κ, *and* $|\mathcal{X}|$ *as constants,* $\tau = \mathcal{O}(\sqrt{\log(1/\gamma)})$ *and hence* $S = \mathcal{O}\left(\frac{(\log(1/\gamma))^2}{\gamma^2} \log\left(\frac{\log(1/\gamma)}{\gamma\delta}\right)\right)$.

Its proof is provided in Appendix B.4.

Remark. It can be observed from Theorem 5 that the number of generated τ-truncated observations has to be increased (i.e., higher chosen value of S) when (a) a lower probability δ of Q_n^ϵ (4.9) approximating Q_n^* (4.7) poorly (i.e., $|Q_n^*(z_\mathcal{D}, x) - Q_n^\epsilon(z_\mathcal{D}, x)| > N\gamma$) is desired, and (b) a larger domain $\mathcal{X}$ of the phenomenon is given. The influence of γ, N, $|\Lambda|$, σ_o, σ_n, and κ on S is similar to that on τ, as previously reported in Remark 2 after Lemma 2.

Thus far, we have shown in the first step that, with high probability, Q_n^ϵ (4.9) approximates Q_n^* (4.7) arbitrarily closely for some chosen values of S and τ (Theorem 5). The next step uses this result to probabilistically bound the performance loss in terms of Q_n^* by observing location $\pi_n^\epsilon(z_\mathcal{D})$ selected by our ϵ-BAL policy π^ϵ at stage n and following the BAL policy π^* thereafter:

Lemma 4. *Suppose that a set* $z_\mathcal{D}$ *of observations, a budget of* $N - n + 1$ *sampling locations for* $1 \leq n \leq N$, $\gamma > 0$, *and* $0 < \delta < 1$ *are given.* $Q_n^*(z_\mathcal{D}, \pi_n^*(z_\mathcal{D})) - Q_n^*(z_\mathcal{D}, \pi_n^\epsilon(z_\mathcal{D})) \leq 2N\gamma$ *holds with probability at least* $1-\delta$ *by setting* S *and* τ *according to that in Theorem 5.*

See Appendix B.5 for its proof. The final step extends Lemma 4 to obtain our second

main result. In particular, it bounds the *expected* active sensing performance loss of our stochastic ϵ-BAL policy π^ϵ relative to that of BAL policy π^*, that is, policy π^ϵ is ϵ-Bayes-optimal:

Theorem 6. *Given a set $z_{\mathcal{D}_0}$ of prior observations, a budget of N sampling locations, and $\epsilon > 0$, $V_1^*(z_{\mathcal{D}_0}) - \mathbb{E}_{\pi^\epsilon}[V_1^{\pi^\epsilon}(z_{\mathcal{D}_0})] \leq \epsilon$ by setting and substituting $\gamma = \epsilon/(4N^2)$ and $\delta = \epsilon/(2N(N\log(\sigma_o/\sigma_n) + \log|\Lambda|))$ into S and τ in Theorem 5 to give $\tau = \mathcal{O}(\sqrt{\log(1/\epsilon)})$ and $S = \mathcal{O}\left(\frac{(\log(1/\epsilon))^2}{\epsilon^2}\log\left(\frac{\log(1/\epsilon)}{\epsilon^2}\right)\right)$.*

Its proof is given in Appendix B.6.

Remark 1. The number of generated τ-truncated observations and the amount of truncation have to be, respectively, increased and reduced (i.e., higher chosen values of S and τ) when a tighter user-defined loss bound ϵ is desired.

Remark 2. The deterministic BAL policy π^* is Bayes-optimal among all candidate stochastic policies π since $\mathbb{E}_\pi[V_1^\pi(z_{\mathcal{D}_0})] \leq V_1^*(z_{\mathcal{D}_0})$, as proven in Appendix B.7.

4.2.4 Anytime ϵ-BAL ($\langle\alpha,\epsilon\rangle$-BAL) Algorithm

Unlike the BAL policy π^*, our ϵ-BAL policy π^ϵ can be derived exactly because its time complexity is independent of the size of the set of all possible originally Gaussian distributed observations, which is uncountable. But, the cost of deriving π^ϵ is exponential in the length N of planning horizon since it has to compute the values $V_n^\epsilon(z_\mathcal{D})$ (4.9) for all $(S|\mathcal{X}|)^N$ possible states $(n, z_\mathcal{D})$. To ease this computational burden, we propose an anytime algorithm based on ϵ-BAL that can produce a good policy fast and improve its approximation quality over time, as discussed next.

The key intuition behind our *anytime* ϵ-BAL *algorithm* ($\langle\alpha,\epsilon\rangle$-BAL of Algo. 1) is to focus the simulation of greedy exploration paths through the most uncertain regions of the state space (i.e., in terms of the values $V_n^\epsilon(z_\mathcal{D})$) instead of evaluating the entire state space like π^ϵ. To achieve this, our $\langle\alpha,\epsilon\rangle$-BAL algorithm maintains both lower and upper heuristic bounds (respectively, $\underline{V}_n^\epsilon(z_\mathcal{D})$ and $\overline{V}_n^\epsilon(z_\mathcal{D})$) for each encountered state $(n, z_\mathcal{D})$, which are exploited for representing the uncertainty of its corresponding value $V_n^\epsilon(z_\mathcal{D})$ to be used in turn for guiding the greedy exploration (or, put differently, pruning unnecessary, bad exploration of the state space while still guaranteeing the policy optimality).

To elaborate, each simulated exploration path (EXPLORE of Algo. 1) repeatedly selects a sampling location x and its corresponding τ-truncated observation z_x^i at every stage until the last stage N is reached. Specifically, at each stage n of the simulated path, the next states $(n+1, z_\mathcal{D}\cup\{z_x^i\})$ with uncertainty $|\overline{V}_{n+1}^\epsilon(z_\mathcal{D}\cup\{z_x^i\})-\underline{V}_{n+1}^\epsilon(z_\mathcal{D}\cup\{z_x^i\})|$ exceeding α (line 6) are identified (lines 7-8), among which the one with largest lower bound $\underline{V}_{n+1}^\epsilon(z_\mathcal{D}\cup\{z_x^i\})$ (line 10) is prioritized/selected for exploration (if more than one exists, ties are broken by choosing the one with most uncertainty, that is, largest upper bound $\overline{V}_{n+1}^\epsilon(z_\mathcal{D}\cup\{z_x^i\})$ (line 11)) while the remaining unexplored ones are placed in the set $\mathcal{U}$ (line 12) to be considered for future exploration (lines 3-6 in $\langle\alpha,\epsilon\rangle$-BAL). So, the simulated path terminates if the uncertainty of every next state is at most α; the uncertainty of a state at the last stage N is guaranteed to be zero (4.15). Then, the algorithm backtracks up the path to update/tighten the bounds of previously visited states (line 7 in $\langle\alpha,\epsilon\rangle$-BAL and line 14 in EXPLORE) as follows:

$$\begin{aligned}\overline{V}_n^\epsilon(z_\mathcal{D}) &\leftarrow \min\left(\overline{V}_n^\epsilon(z_\mathcal{D}), \max_{x\in\mathcal{X}\setminus\mathcal{D}} \overline{Q}_n^\epsilon(z_\mathcal{D},x)\right)\\ \underline{V}_n^\epsilon(z_\mathcal{D}) &\leftarrow \max\left(\underline{V}_n^\epsilon(z_\mathcal{D}), \max_{x\in\mathcal{X}\setminus\mathcal{D}} \underline{Q}_n^\epsilon(z_\mathcal{D},x)\right)\end{aligned} \tag{4.15}$$

where we define $\overline{Q}_n^{\epsilon}(z_{\mathcal{D}}, x)$ and $\underline{Q}_n^{\epsilon}(z_{\mathcal{D}}, x)$ as

$$\begin{aligned}
\overline{Q}_n^{\epsilon}(z_{\mathcal{D}}, x) &\triangleq \frac{1}{S}\sum_{i=1}^{S}\left(-\log p\left(z_x^i | z_{\mathcal{D}}\right) + \overline{V}_{n+1}^{\epsilon}\big(z_{\mathcal{D}} \cup \left\{z_x^i\right\}\big)\right) \\
\underline{Q}_n^{\epsilon}(z_{\mathcal{D}}, x) &\triangleq \frac{1}{S}\sum_{i=1}^{S}\left(-\log p\left(z_x^i | z_{\mathcal{D}}\right) + \underline{V}_{n+1}^{\epsilon}\left(z_{\mathcal{D}} \cup \left\{z_x^i\right\}\right)\right)
\end{aligned}$$

for $n = 1, \ldots, N$ such that there is no $\overline{V}_{N+1}^{\epsilon}$ ($\underline{V}_{N+1}^{\epsilon}$) term on the RHS of $\overline{Q}_N^{\epsilon}$ ($\underline{Q}_N^{\epsilon}$) at stage $n = N$. When the planning time runs out, we provide the greedy policy induced by the lower bound: $\pi_1^{\langle\alpha,\epsilon\rangle}(z_{\mathcal{D}_0}) \triangleq \arg\max_{x \in \mathcal{X}\setminus\mathcal{D}_0} \underline{Q}_1^{\epsilon}(z_{\mathcal{D}_0}, x)$ (line 8 in $\langle\alpha,\epsilon\rangle$-BAL).

Central to the anytime performance of our $\langle\alpha,\epsilon\rangle$-BAL algorithm is the computational efficiency of deriving informed initial heuristic bounds $\underline{V}_n^{\epsilon}(z_{\mathcal{D}})$ and $\overline{V}_n^{\epsilon}(z_{\mathcal{D}})$ where $\underline{V}_n^{\epsilon}(z_{\mathcal{D}}) \leq V_n^{\epsilon}(z_{\mathcal{D}}) \leq \overline{V}_n^{\epsilon}(z_{\mathcal{D}})$. Due to the use of the truncated sampling procedure (Section 4.2.2), computing informed initial heuristic bounds for $V_n^{\epsilon}(z_{\mathcal{D}})$ is infeasible without expanding from its corresponding state to all possible states in the subsequent stages $n+1, \ldots, N$, which we want to avoid. To resolve this issue, we instead derive informed bounds $\underline{V}_n^{\epsilon}(z_{\mathcal{D}})$ and $\overline{V}_n^{\epsilon}(z_{\mathcal{D}})$ that satisfy

$$\underline{V}_n^{\epsilon}(z_{\mathcal{D}}) \;\leq\; V_n^{\epsilon}(z_{\mathcal{D}}) \;\leq\; \overline{V}_n^{\epsilon}(z_{\mathcal{D}})\,. \tag{4.16}$$

with high probability: Using Theorem 1, $|V_n^*(z_{\mathcal{D}}) - V_n^{\epsilon}(z_{\mathcal{D}})| \leq \max_{x\in\mathcal{X}\setminus\mathcal{D}} |Q_n^*(z_{\mathcal{D}}, x) - Q_n^{\epsilon}(z_{\mathcal{D}}, x)| \leq N\gamma$, which implies $V_n^*(z_{\mathcal{D}}) - N\gamma \leq V_n^{\epsilon}(z_{\mathcal{D}}) \leq V_n^*(z_{\mathcal{D}}) + N\gamma$ with probability at least $1-\delta$. $V_n^*(z_{\mathcal{D}})$ can at least be naively bounded using the uninformed, domain-independent lower and upper bounds given in Lemma 11. In practice, domain-dependent bounds $\underline{V}_n^*(z_{\mathcal{D}})$ and $\overline{V}_n^*(z_{\mathcal{D}})$ (i.e., $\underline{V}_n^*(z_{\mathcal{D}}) \leq V_n^*(z_{\mathcal{D}}) \leq \overline{V}_n^*(z_{\mathcal{D}})$) tend to be more informed and we will show in Theorem 7 below how they can be derived efficiently. So, by setting $\underline{V}_n^{\epsilon}(z_{\mathcal{D}}) = \underline{V}_n^*(z_{\mathcal{D}}) - N\gamma$ and $\overline{V}_n^{\epsilon}(z_{\mathcal{D}}) = \overline{V}_n^*(z_{\mathcal{D}}) + N\gamma$

Algorithm 1 $\langle\alpha,\epsilon\rangle$-BAL$(z_{\mathcal{D}_0})$

$\langle\alpha,\epsilon\rangle$-BAL$(z_{\mathcal{D}_0})$

$\mathcal{U} \leftarrow \{(1, z_{\mathcal{D}_0})\}$
while $|\overline{V}_1^{\epsilon}(z_{\mathcal{D}_0}) - \underline{V}_1^{\epsilon}(z_{\mathcal{D}_0})| > \alpha$ **do**
  $\mathcal{V} \leftarrow \arg\max_{(n,z_{\mathcal{D}})\in\mathcal{U}} \underline{V}_n^{\epsilon}(z_{\mathcal{D}})$
  $(n', z_{\mathcal{D}'}) \leftarrow \arg\max_{(n,z_{\mathcal{D}})\in\mathcal{V}} \overline{V}_n^{\epsilon}(z_{\mathcal{D}})$
  $\mathcal{U} \leftarrow \mathcal{U} \setminus \{(n', z_{\mathcal{D}'})\}$
  EXPLORE$(n', z_{\mathcal{D}'}, \mathcal{U})$ /* $\mathcal{U}$ is passed by reference */
  UPDATE$(n', z_{\mathcal{D}'})$
return $\pi_1^{\langle\alpha,\epsilon\rangle}(z_{\mathcal{D}_0}) \leftarrow \arg\max_{x\in\mathcal{X}\setminus\mathcal{D}_0} \underline{Q}_1^{\epsilon}(z_{\mathcal{D}_0}, x)$

EXPLORE$(n, z_{\mathcal{D}}, \mathcal{U})$

$\mathcal{T} \leftarrow \emptyset$
for all $x \in \mathcal{X} \setminus \mathcal{D}$ **do**
  $\{z_x^i\}_{i=1}^S \leftarrow$ sample from $p(z_x|z_{\mathcal{D}})$ (4.5)
  for $i = 1, \ldots, S$ **do**
    $z_x^i \leftarrow z_x^i \min(|z_x^i|, \widehat{\tau}) / |z_x^i|$
    if $|\overline{V}_{n+1}^{\epsilon}(z_{\mathcal{D}} \cup \{z_x^i\}) - \underline{V}_{n+1}^{\epsilon}(z_{\mathcal{D}} \cup \{z_x^i\})| > \alpha$ **then**
      $\mathcal{T} \leftarrow \mathcal{T} \cup \{(n+1, z_{\mathcal{D}} \cup \{z_x^i\})\}$
      parent$(n+1, z_{\mathcal{D}} \cup \{z_x^i\}) \leftarrow (n, z_{\mathcal{D}})$
if $|\mathcal{T}| > 0$ **then**
  $\mathcal{V} \leftarrow \arg\max_{(n+1,z_{\mathcal{D}}\cup\{z_x^i\})\in\mathcal{T}} \underline{V}_{n+1}^{\epsilon}(z_{\mathcal{D}} \cup \{z_x^i\})$
  $(n+1, z_{\mathcal{D}'}) \leftarrow \arg\max_{(n+1,z_{\mathcal{D}}\cup\{z_x^i\})\in\mathcal{V}} \overline{V}_{n+1}^{\epsilon}(z_{\mathcal{D}} \cup \{z_x^i\})$
  $\mathcal{U} \leftarrow \mathcal{U} \cup (\mathcal{T} \setminus \{(n+1, z_{\mathcal{D}'})\})$
  EXPLORE$(n+1, z_{\mathcal{D}'}, \mathcal{U})$
Update $\overline{V}_n^{\epsilon}(z_{\mathcal{D}})$ and $\underline{V}_n^{\epsilon}(z_{\mathcal{D}})$ using (4.15)

UPDATE$(n, z_{\mathcal{D}})$

Update $\overline{V}_n^{\epsilon}(z_{\mathcal{D}})$ and $\underline{V}_n^{\epsilon}(z_{\mathcal{D}})$ using (4.15)
if $n > 1$ **then**
  $(n-1, z_{\mathcal{D}'}) \leftarrow$ parent$(n, z_{\mathcal{D}})$
  UPDATE$(n-1, z_{\mathcal{D}'})$

for $n < N$ and $\underline{V}_N^{\epsilon}(z_{\mathcal{D}}) = \overline{V}_N^{\epsilon}(z_{\mathcal{D}}) = \max_{x\in\mathcal{X}\setminus\mathcal{D}} S^{-1} \sum_{i=1}^S -\log p(z_x^i|z_{\mathcal{D}})$, (4.16) holds with probability at least $1-\delta$.

Theorem 7. *Given a set $z_{\mathcal{D}}$ of observations and a space Λ of parameters λ, define the a priori greedy design with unknown parameters as the set $\mathcal{S}_n$ of $n \geq 1$ sampling*

locations where

$$\begin{aligned}\mathcal{S}_0 &\triangleq \emptyset \\ \mathcal{S}_n &\triangleq \mathcal{S}_{n-1} \cup \Bigg\{ \arg\max_{x\in\mathcal{X}} \sum_{\lambda\in\Lambda} b_{\mathcal{D}}(\lambda)\ \mathbb{H}\left[Z_{\mathcal{S}_{n-1}\cup\{x\}}|z_{\mathcal{D}},\lambda\right] \\ &\qquad\qquad - \sum_{\lambda\in\Lambda} b_{\mathcal{D}}(\lambda)\ \mathbb{H}\left[Z_{\mathcal{S}_{n-1}}|z_{\mathcal{D}},\lambda\right] \Bigg\}.\end{aligned} \tag{4.17}$$

Similarly, define the a priori greedy design with known parameters λ as the set $\mathcal{S}_n^\lambda$ of $n \geq 1$ sampling locations where

$$\begin{aligned}\mathcal{S}_0^\lambda &\triangleq \emptyset \\ \mathcal{S}_n^\lambda &\triangleq \mathcal{S}_{n-1}^\lambda \cup \Big\{ \arg\max_{x\in\mathcal{X}}\ \mathbb{H}\left[Z_{\mathcal{S}_{n-1}^\lambda\cup\{x\}}|z_{\mathcal{D}},\lambda\right] \\ &\qquad\qquad - \mathbb{H}\left[Z_{S_{n-1}^\lambda}|z_{\mathcal{D}},\lambda\right] \Big\}.\end{aligned} \tag{4.18}$$

Then, it follows that

$$\begin{aligned}\mathbb{H}\left[Z_{\{\pi_i^*\}_{i=N-n+1}^N}|z_{\mathcal{D}}\right] &\geq \sum_{\lambda\in\Lambda} b_{\mathcal{D}}(\lambda)\ \mathbb{H}\left[Z_{\mathcal{S}_n}|z_{\mathcal{D}},\lambda\right] \\ \mathbb{H}\left[Z_{\{\pi_i^*\}_{i=N-n+1}^N}|z_{\mathcal{D}}\right] &\leq \sum_{\lambda\in\Lambda} b_{\mathcal{D}}(\lambda)\left[\frac{e}{e-1}\mathbb{H}\left[Z_{\mathcal{S}_n^\lambda}|z_{\mathcal{D}},\lambda\right] + \frac{nr}{e-1}\right] \\ &\quad + \mathbb{H}\left[\Lambda\right]\end{aligned} \tag{4.19}$$

where

$$\{\pi_i^*\}_{i=N-n+1}^N = \arg\max_{\{\pi_i\}_{i=N-n+1}^N}\ \mathbb{H}\left[Z_{\{\pi_i\}_{i=N-n+1}^N}|z_{\mathcal{D}}\right]\ ,$$

Λ denotes the set of random parameters corresponding to the realized parameters λ, and $r = -\min(0, 0.5\log(2\pi e\sigma_n^2)) \geq 0$.

Remark. $V_{N-n+1}^*(z_{\mathcal{D}}) = \mathbb{H}[Z_{\{\pi_i^*\}_{i=N-n+1}^N}|z_{\mathcal{D}}]$, by definition. Hence, the lower and upper bounds of $\mathbb{H}[Z_{\{\pi_i^*\}_{i=N-n+1}^N}|z_{\mathcal{D}}]$ (4.19) constitute informed domain-dependent

bounds for $V^*_{N-n+1}(z_{\mathcal{D}})$ that can be derived efficiently since both $\mathcal{S}_n$ (4.17) and $\{\mathcal{S}_n^{\lambda}\}_{\lambda\in\Lambda}$ (4.18) can be computed in polynomial time with respect to the interested variables.

Proof. To prove the lower bound,

$$\begin{aligned}
\mathbb{H}\left[Z_{\{\pi_i^*\}_{i=N-n+1}^N}|z_{\mathcal{D}}\right] &= \max_{\{\pi_i\}_{i=N-n+1}^N} \mathbb{H}\left[Z_{\{\pi_i\}_{i=N-n+1}^N}|z_{\mathcal{D}}\right] \\
&\geq \max_{\{\pi_i\}_{i=N-n+1}^N} \sum_{\lambda\in\Lambda} b_{\mathcal{D}}(\lambda)\ \mathbb{H}\left[Z_{\{\pi_i\}_{i=N-n+1}^N}|z_{\mathcal{D}},\lambda\right] \\
&\geq \max_{\mathcal{S}\subseteq\mathcal{X}:|\mathcal{S}|=n} \sum_{\lambda\in\Lambda} b_{\mathcal{D}}(\lambda)\ \mathbb{H}\left[Z_{\mathcal{S}}|z_{\mathcal{D}},\lambda\right] \\
&\geq \sum_{\lambda\in\Lambda} b_{\mathcal{D}}(\lambda)\ \mathbb{H}\left[Z_{\mathcal{S}_n}|z_{\mathcal{D}},\lambda\right]\ .
\end{aligned}$$

The first inequality follows from the monotonicity of conditional entropy (i.e., "information never hurts" bound) [Cover and Thomas, 1991]. The second inequality holds because the optimal set $\mathcal{S}^* \triangleq \arg\max_{\mathcal{S}\subseteq\mathcal{X}:|\mathcal{S}|=n} \sum_{\lambda\in\Lambda} b_{\mathcal{D}}(\lambda)\ \mathbb{H}\left[Z_{\mathcal{S}}|z_{\mathcal{D}},\lambda\right]$ is an optimal *a priori* design (i.e., non-sequential) that does not perform better than the optimal sequential policy π^* [Krause and Guestrin, 2007]. The third inequality is due to definition of $\mathcal{S}_n$.

To prove the upper bound,

$$\begin{aligned}
\mathbb{H}\left[Z_{\{\pi_i^*\}_{i=N-n+1}^N}|z_{\mathcal{D}}\right] &\leq \sum_{\lambda\in\Lambda} b_{\mathcal{D}}(\lambda) \max_{\mathcal{S}\subseteq\mathcal{X}:|\mathcal{S}|=n} \mathbb{H}\left[Z_{\mathcal{S}}|z_{\mathcal{D}},\lambda\right] + \mathbb{H}\left[\Lambda\right] \\
&\leq \sum_{\lambda\in\Lambda} b_{\mathcal{D}}(\lambda) \left[\frac{e}{e-1}\mathbb{H}\left[Z_{\mathcal{S}_n^{\lambda}}|z_{\mathcal{D}},\lambda\right] + \frac{nr}{e-1}\right] + \mathbb{H}\left[\Lambda\right]
\end{aligned}$$

such that the first inequality is due to Theorem 1 of Krause and Guestrin [2007], and the second inequality follows from Lemma 19. □

Finally, we provide a theoretical guarantee similar to that of Theorem 6 on the expected active sensing performance of our $\langle\alpha,\epsilon\rangle$-BAL policy $\pi^{\langle\alpha,\epsilon\rangle}$ (Section 4.2.5) and analyze the time complexity of simulating k exploration paths in our $\langle\alpha,\epsilon\rangle$-BAL algorithm (Section 4.2.6) to conclude this section.

4.2.5 Performance Guarantee of $\langle\alpha,\epsilon\rangle$-BAL Policy $\pi^{\langle\alpha,\epsilon\rangle}$

Lemma 5. *Suppose that a set $z_{\mathcal{D}}$ of observations, a budget of $N-n+1$ sampling locations for $1\leq n\leq N$, $\gamma>0$, $0<\delta<1$, and $\alpha>0$ are given. $Q_n^*(z_{\mathcal{D}},\pi_n^*(z_{\mathcal{D}}))-Q_n^*(z_{\mathcal{D}},\pi_n^{\langle\alpha,\epsilon\rangle}(z_{\mathcal{D}}))\leq 2(N\gamma+\alpha)$ holds with probability at least $1-\delta$ by setting S and τ according to that in Theorem 5.*

Proof. When our $\langle\alpha,\epsilon\rangle$-BAL algorithm terminates, $|\overline{V}_1^\epsilon(z_{\mathcal{D}_0})-\underline{V}_1^\epsilon(z_{\mathcal{D}_0})|\leq\alpha$, which implies $|V_1^\epsilon(z_{\mathcal{D}_0})-\underline{V}_1^\epsilon(z_{\mathcal{D}_0})|\leq\alpha$. By Theorem 1, since $|V_1^*(z_{\mathcal{D}_0})-V_1^\epsilon(z_{\mathcal{D}_0})|\leq\max_{x\in\mathcal{X}\setminus\mathcal{D}_0}|Q_n^*(z_{\mathcal{D}_0},x)-Q_n^\epsilon(z_{\mathcal{D}_0},x)|\leq N\gamma$, $|V_1^*(z_{\mathcal{D}_0})-\underline{V}_1^\epsilon(z_{\mathcal{D}_0})|\leq|V_1^*(z_{\mathcal{D}_0})-V_1^\epsilon(z_{\mathcal{D}_0})|+|V_1^\epsilon(z_{\mathcal{D}_0})-\underline{V}_1^\epsilon(z_{\mathcal{D}_0})|\leq N\gamma+\alpha$ with probability at least $1-\delta$. In general, given that the length of planning horizon is reduced to $N-n+1$ for $1\leq n\leq N$, the above inequalities are equivalent to

$$\begin{aligned}|V_n^\epsilon(z_{\mathcal{D}})-\underline{V}_n^\epsilon(z_{\mathcal{D}})| &\leq \alpha\\ |V_n^*(z_{\mathcal{D}})-\underline{V}_n^\epsilon(z_{\mathcal{D}})| &= \left|Q_n^*(z_{\mathcal{D}_0},\pi_n^*(z_{\mathcal{D}}))-\underline{Q}_n^\epsilon(z_{\mathcal{D}},\pi_n^{\langle\alpha,\epsilon\rangle}(z_{\mathcal{D}}))\right|\\ &\leq N\gamma+\alpha \qquad (4.20)\end{aligned}$$

by increasing/shifting the indices of V_1^ϵ, $\underline{V}_1^\epsilon$, and V_1^* above from 1 to n so that these

value functions start at stage n instead.

$$\begin{aligned}
Q_n^*(z_{\mathcal{D}}, \pi_n^*(z_{\mathcal{D}})) - Q_n^*(z_{\mathcal{D}}, \pi_n^{\langle\alpha,\epsilon\rangle}(z_{\mathcal{D}})) &= Q_n^*(z_{\mathcal{D}}, \pi_n^*(z_{\mathcal{D}})) - \underline{Q}_n^{\epsilon}(z_{\mathcal{D}}, \pi_n^{\langle\alpha,\epsilon\rangle}(z_{\mathcal{D}})) \\
&+ \underline{Q}_n^{\epsilon}(z_{\mathcal{D}}, \pi_n^{\langle\alpha,\epsilon\rangle}(z_{\mathcal{D}})) - Q_n^{\epsilon}(z_{\mathcal{D}}, \pi_n^{\langle\alpha,\epsilon\rangle}(z_{\mathcal{D}})) \\
&+ Q_n^{\epsilon}(z_{\mathcal{D}}, \pi_n^{\langle\alpha,\epsilon\rangle}(z_{\mathcal{D}})) - Q_n^*(z_{\mathcal{D}}, \pi_n^{\langle\alpha,\epsilon\rangle}(z_{\mathcal{D}})) \\
&\leq N\gamma + \alpha + \frac{1}{S}\sum_{i=1}^{S}\Big(\underline{V}_{n+1}^{\epsilon}(z_{\mathcal{D}} \cup \{z^i_{\pi_n^{\langle\alpha,\epsilon\rangle}(z_{\mathcal{D}})}\}) \\
&- V_{n+1}^{\epsilon}(z_{\mathcal{D}} \cup \{z^i_{\pi_n^{\langle\alpha,\epsilon\rangle}(z_{\mathcal{D}})}\})\Big) + N\gamma \\
&\leq 2(N\gamma + \alpha)
\end{aligned}$$

where the inequalities follow from (4.9), (4.15), (4.20), and Theorem 1. □

Theorem 8. *Given a set $z_{\mathcal{D}_0}$ of prior observations, a budget of N sampling locations, $\alpha > 0$, and $\epsilon > 4N\alpha$, $V_1^*(z_{\mathcal{D}_0}) - \mathbb{E}_{\pi^{\langle\alpha,\epsilon\rangle}}\left[V_1^{\pi^{\langle\alpha,\epsilon\rangle}}(z_{\mathcal{D}_0})\right] \leq \epsilon$ by setting and substituting $\gamma = \epsilon/(4N^2)$ and $\delta = (\epsilon/(2N) - 2\alpha)/(N\log(\sigma_o/\sigma_n) + \log|\Lambda|)$ into S and τ in Theorem 5 to give $\tau = \mathcal{O}(\sqrt{\log(1/\epsilon)})$ and $S = \mathcal{O}\left(\frac{(\log(1/\epsilon))^2}{\epsilon^2}\log\left(\frac{\log(1/\epsilon)}{\epsilon(\epsilon-\alpha)}\right)\right)$.*

Proof Sketch. The proof directly follows from Lemma 5 and is similar to that of Theorem 6. □

4.2.6 Time Complexity of $\langle\alpha,\epsilon\rangle$-BAL Algorithm

Suppose that our $\langle\alpha,\epsilon\rangle$-BAL algorithm runs k simulated exploration paths during its lifetime where k actually depends on the available time for planning. Then, since each exploration path has at most N stages and each stage generates at most $S|\mathcal{X}|$ states, there will be at most $\mathcal{O}(kNS|\mathcal{X}|)$ states generated during the whole lifetime of our $\langle\alpha,\epsilon\rangle$-BAL algorithm. So, to analyze the overall time complexity of our $\langle\alpha,\epsilon\rangle$-BAL algorithm, the processing cost at each state is first quantified, which, according to EXPLORE of Algorithm 1, includes the cost of sampling (lines 2-5), initializing (line

6) and updating the corresponding heuristic bounds (line 14). In particular, the cost of sampling at each state involves training the GPs (i.e., $\mathcal{O}(N^3)$) and computing the predictive distributions using (4.2) and (4.3) (i.e., $\mathcal{O}(|\mathcal{X}|N^2)$) for each set of realized parameters $\lambda \in \Lambda$ and the cost of generating $S|\mathcal{X}|$ samples from a mixture of $|\Lambda|$ Gaussian distributions (i.e., $\mathcal{O}(|\Lambda|S|\mathcal{X}|)$) by assuming that drawing a sample from a Gaussian distribution consumes a unit processing cost. This results in a total sampling complexity of $\mathcal{O}(|\Lambda|(N^3 + |\mathcal{X}|N^2 + S|\mathcal{X}|))$.

Now, let $\mathcal{O}(\Delta)$ denote the processing cost of initializing the heuristic bounds at each state, which depends on the actual bounding scheme being used. The total processing cost at each state is therefore $\mathcal{O}(|\Lambda|(N^3 + |\mathcal{X}|N^2 + S|\mathcal{X}|) + \Delta + S|\mathcal{X}|)$ where the last term corresponds to the cost of updating bounds by (4.15). In addition, to search for the most potential state to explore in $\mathcal{O}(1)$ at each stage (lines 10-11), the set of unexplored states is maintained in a priority queue (line 12) using the corresponding exploration criterion, thus incurring an extra management cost (i.e., updating the queue) of $\mathcal{O}(\log(kNS|\mathcal{X}|))$. That is, the total time complexity of simulating k exploration paths in our $\langle\alpha, \epsilon\rangle$-BAL algorithm is $\mathcal{O}(kNS|\mathcal{X}|(|\Lambda|(N^3 + |\mathcal{X}|N^2 + S|\mathcal{X}|) + \Delta + \log(kNS|\mathcal{X}|)))$. In practice, $\langle\alpha, \epsilon\rangle$-BAL's planning horizon can be shortened to reduce its computational cost further by limiting the depth of each simulated path to strictly less than N. In that case, although the resulting $\pi^{\langle\alpha,\epsilon\rangle}$'s performance has not been theoretically analyzed, Section 4.3 demonstrates empirically that it outperforms state-of-the-art algorithms.

4.3 Experiments and Discussion

This section evaluates the active sensing performance and time efficiency of our $\langle\alpha, \epsilon\rangle$-BAL policy $\pi^{\langle\alpha,\epsilon\rangle}$ (Section 4.2) empirically under limited sampling budget using two

datasets featuring a simple, simulated spatial phenomenon (Section 4.3.1) and a large-scale, real-world traffic phenomenon (i.e., speeds of road segments) over an urban road network (Section 4.3.2). All experiments are run on a Mac OS X machine with Intel Core i7 at 2.66 GHz.

4.3.1 Simulated Spatial Phenomenon

The domain of the phenomenon is discretized into a finite set of sampling locations $\mathcal{X} = \{0, 1, \ldots, 99\}$. The phenomenon is a realization of a GP (Section 4.1) parameterized by $\lambda^* = \{\sigma_n^{\lambda^*} = 0.25, \sigma_s^{\lambda^*} = 10.0, \ell^{\lambda^*} = 1.0\}$. For simplicity, we assume that $\sigma_n^{\lambda^*}$ and $\sigma_s^{\lambda^*}$ are known, but the true length-scale $\ell^{\lambda^*} = 1$ is not. So, a uniform prior belief $b_{\mathcal{D}_0=\emptyset}$ is maintained over a set $\mathcal{L} = \{1, 6, 9, 12, 15, 18, 21\}$ of 7 candidate length-scales ℓ^{λ}. Using *root mean squared prediction error* (RMSPE) as the performance metric, the performance of our $\langle\alpha, \epsilon\rangle$-BAL policies $\pi^{\langle\alpha,\epsilon\rangle}$ with planning horizon length $N' = 2, 3$ and $\alpha = 1.0$ are compared to that of the state-of-the-art GP-based active learning algorithms: (a) The *a priori greedy design* (APGD) policy [Shewry and Wynn, 1987] iteratively selects and adds $\arg\max_{x\in\mathcal{X}\setminus\mathcal{S}_n} \sum_{\lambda\in\Lambda} b_{\mathcal{D}_0}(\lambda)\mathbb{H}[Z_{\mathcal{S}_n\cup\{x\}}|z_{\mathcal{D}_0}, \lambda]$ to the current set $\mathcal{S}_n$ of sampling locations (where $\mathcal{S}_0 = \emptyset$) until $\mathcal{S}_N$ is obtained, (b) the *implicit exploration* (IE) policy greedily selects and observes sampling location $x^{\mathrm{IE}} = \arg\max_{x\in\mathcal{X}\setminus\mathcal{D}} \sum_{\lambda\in\Lambda} b_{\mathcal{D}}(\lambda)\mathbb{H}[Z_x|z_{\mathcal{D}}, \lambda]$ and updates the belief from $b_{\mathcal{D}}$ to $b_{\mathcal{D}\cup\{x^{\mathrm{IE}}\}}$ over $\mathcal{L}$; if the upper bound on the performance advantage of using π^* over APGD policy is less than a pre-defined threshold, it will use APGD with the remaining sampling budget, and (c) the *explicit exploration via independent tests* (ITE) policy performs a PAC-based binary search, which is guaranteed to find ℓ^{λ^*} with high probability, and then uses APGD to select the remaining locations to be observed.

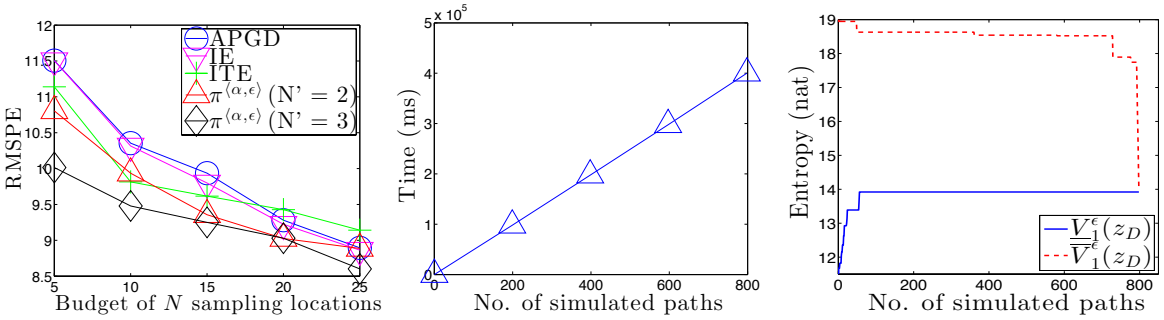


Figure 4.1: Graphs of (a) RMSPE of APGD, IE, ITE, and $\langle\alpha,\epsilon\rangle$-BAL policies with planning horizon length $N' = 2, 3$ vs. budget of N sampling locations, (b) stage-wise online processing cost of $\langle\alpha,\epsilon\rangle$-BAL policy with $N' = 3$ and (c) gap between the heuristic upper- and lower-bounds of $V_1^{\epsilon}(z_{\mathcal{D}_0})$ vs. number of simulated paths.

Both nonmyopic IE and ITE policies are proposed by Krause and Guestrin [2007]: IE is reported to incur the lowest prediction error empirically while ITE is guaranteed not to achieve worse than the optimal performance by more than a factor of $1/e$. Fig. 4.1a shows results of the active sensing performance of the tested policies averaged over 20 realizations of the phenomenon drawn independently from the underlying GP model described earlier. It can be observed that the RMSPE of every tested policy decreases with a larger budget of N sampling locations. Notably, our $\langle\alpha,\epsilon\rangle$-BAL policies perform better than the APGD, IE, and ITE policies, especially when N is small. The performance gap between our $\langle\alpha,\epsilon\rangle$-BAL policies and the other policies decreases as N increases, which intuitively means that, with a tighter sampling budget (i.e., smaller N), it is more critical to strike a right balance between exploration and exploitation.

Fig. 4.2 shows the stage-wise sampling designs produced by the tested policies with a budget of $N = 15$ sampling locations. It can be observed that our $\langle\alpha,\epsilon\rangle$-BAL policy achieves a better balance between exploration and exploitation and can therefore

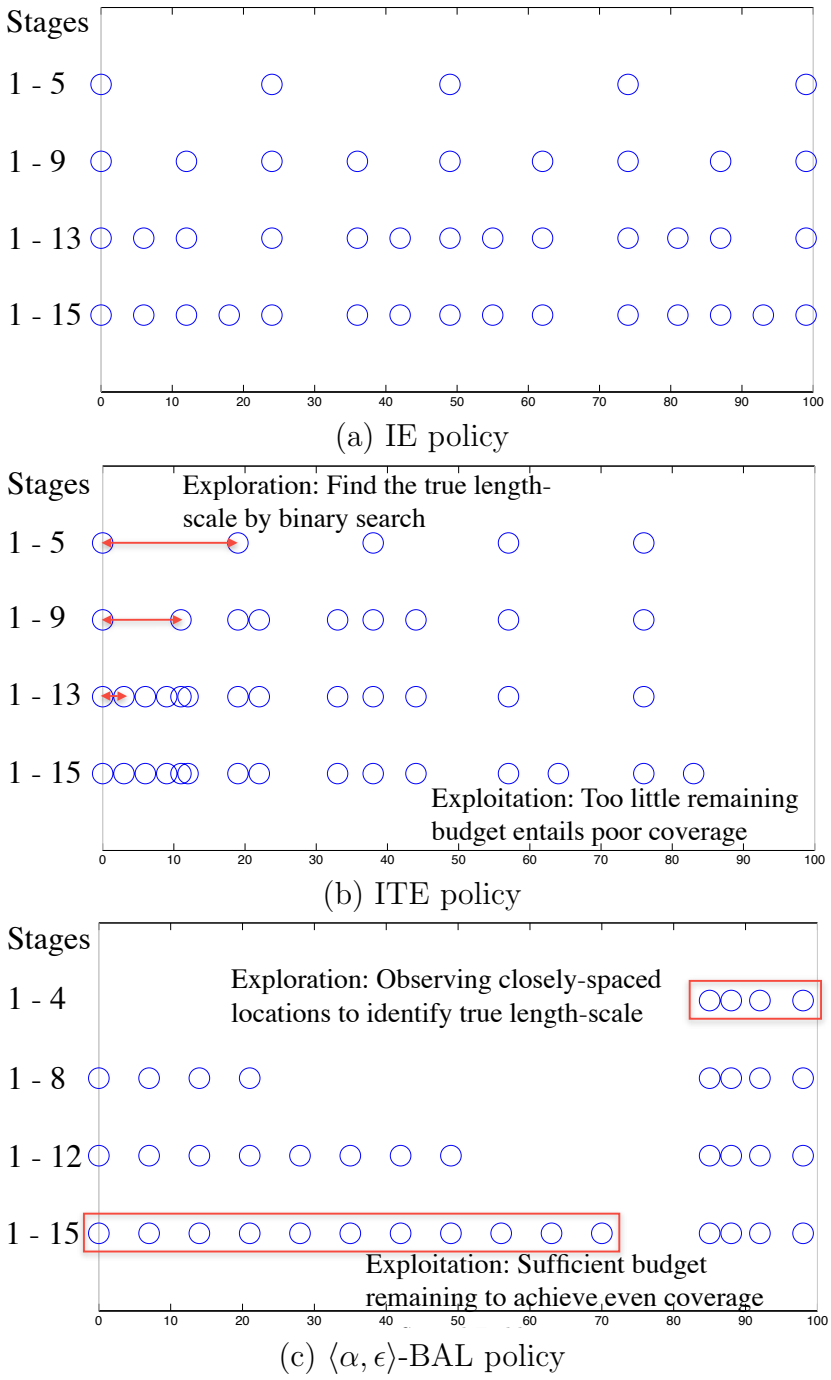


Figure 4.2: Stage-wise sampling designs produced by (a) IE, (b) ITE, and (c) $\langle \alpha, \epsilon \rangle$-BAL policy with a planning horizon length $N' = 3$ using a budget of $N = 15$ sampling locations. The final sampling designs are depicted in the bottommost rows of the figures.

discern ℓ^{λ^*} much faster than the IE and ITE policies while maintaining a fine spatial coverage of the phenomenon. This is expected due to the following issues faced by IE and ITE policies: (a) The myopic exploration of IE tends not to observe closely-spaced locations (Fig. 4.2a), which are in fact informative towards estimating the true length-scale, and (b) despite ITE's theoretical guarantee in finding ℓ^{λ^*}, its PAC-style exploration is too aggressive, hence completely ignoring how informative the posterior belief $b_{\mathcal{D}}$ over $\mathcal{L}$ is during exploration. This leads to a sub-optimal exploration behavior that reserves too little budget for exploitation and consequently entails a poor spatial coverage, as shown in Fig. 4.2b.

Our $\langle \alpha, \epsilon \rangle$-BAL policy can resolve these issues by jointly and naturally optimizing the trade-off between observing the most informative locations for minimizing the predictive uncertainty of the phenomenon (i.e., exploitation) vs. the uncertainty surrounding its length-scale (i.e., exploration), hence enjoying the best of both worlds (Fig. 4.2c). In fact, we notice that, after observing 5 locations, our $\langle \alpha, \epsilon \rangle$-BAL policy can focus 88.10% of its posterior belief on ℓ^{λ^*} while IE only assigns, on average, about 18.65% of its posterior belief on ℓ^{λ^*}, which is hardly more informative than the prior belief $b_{\mathcal{D}_0}(\ell^{\lambda^*}) = 1/7 \approx 14.28\%$. Finally, Fig. 4.1b shows that the online processing cost of $\langle \alpha, \epsilon \rangle$-BAL per sampling stage grows linearly in the number of simulated paths while Fig. 4.1c reveals that its approximation quality improves (i.e., gap between $\overline{V}_1^{\epsilon}(z_{\mathcal{D}_0})$ and $\underline{V}_1^{\epsilon}(z_{\mathcal{D}_0})$ decreases) with increasing number of simulated paths. Interestingly, it can be observed from Fig. 4.1c that although $\langle \alpha, \epsilon \rangle$-BAL needs about 800 simulated paths (i.e., 400 s) to close the gap between $\overline{V}_1^{\epsilon}(z_{\mathcal{D}_0})$ and $\underline{V}_1^{\epsilon}(z_{\mathcal{D}_0})$, $\underline{V}_1^{\epsilon}(z_{\mathcal{D}_0})$ only takes about 100 simulated paths (i.e., 50 s). This implies the actual computation time needed for $\langle \alpha, \epsilon \rangle$-BAL to reach $V_1^{\epsilon}(z_{\mathcal{D}_0})$ (via its lower bound $\underline{V}_1^{\epsilon}(z_{\mathcal{D}_0})$) is much less than that required to verify the convergence of $\underline{V}_1^{\epsilon}(z_{\mathcal{D}_0})$ to $V_1^{\epsilon}(z_{\mathcal{D}_0})$ (i.e., by checking the gap). This is expected since $\langle \alpha, \epsilon \rangle$-BAL explores states

with largest lower bound first (Section 4.2.4).

4.3.2 Real-World Traffic Phenomenon

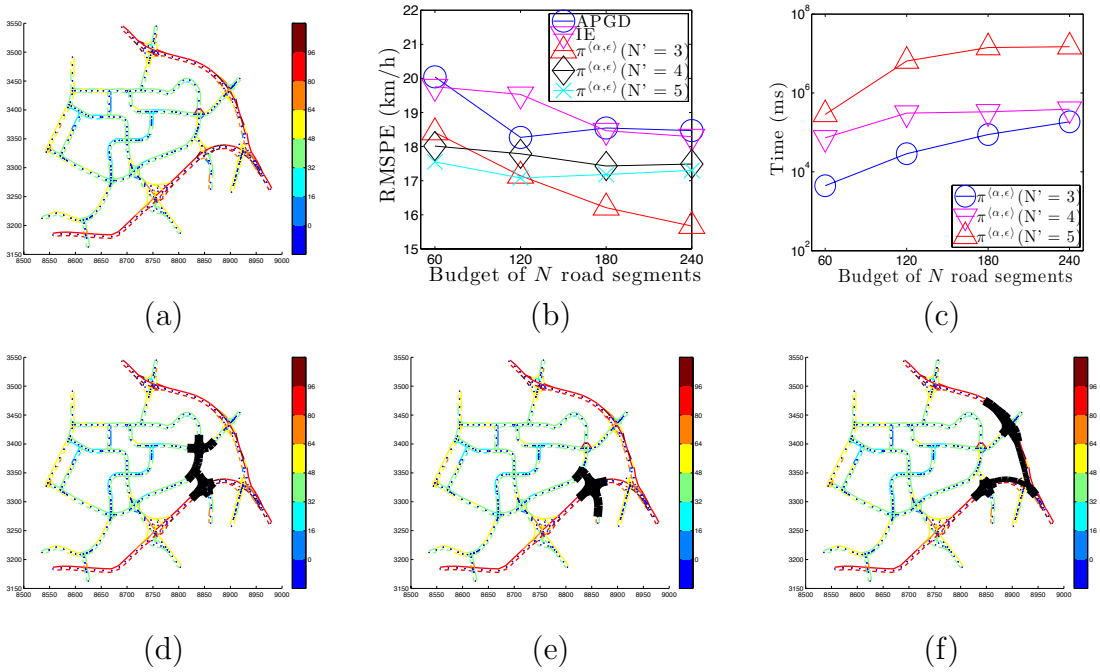


Figure 4.3: (a) Traffic phenomenon (i.e., speeds (km/h) of road segments) over an urban road network in Tampines area, Singapore, graphs of (b) RMSPE of APGD, IE, and $\langle\alpha,\epsilon\rangle$-BAL policies with horizon length $N' = 3, 4, 5$ and (c) total online processing cost of $\langle\alpha,\epsilon\rangle$-BAL policies with $N' = 3, 4, 5$ vs. budget of N segments, and (d-f) road segments observed (shaded in black) by respective APGD, IE, and $\langle\alpha,\epsilon\rangle$-BAL policies ($N' = 5$) with $N = 60$.

Fig. 4.3a shows the traffic phenomenon (i.e., speeds (km/h) of road segments) over an urban road network $\mathcal{X}$ comprising 775 road segments (e.g., highways, arterials, slip roads, etc.) in Tampines area, Singapore during lunch hours on June 20, 2011. The mean speed is 52.8 km/h and the standard deviation is 21.0 km/h. Each road segment $x \in \mathcal{X}$ is specified by a 4-dimensional vector of features: length, number of lanes, speed limit, and direction. The phenomenon is modeled as a relational GP [Chen *et*

al., 2012] whose correlation structure can exploit both the road segment features and road network topology information. The true parameters $\lambda^* = \{\sigma_n^{\lambda^*}, \sigma_s^{\lambda^*}, \ell^{\lambda^*}\}$ are set as the maximum likelihood estimates learned using the entire dataset. We assume that $\sigma_n^{\lambda^*}$ and $\sigma_s^{\lambda^*}$ are known, but ℓ^{λ^*} is not. So, a uniform prior belief $b_{\mathcal{D}_0=\emptyset}$ is maintained over a set $\mathcal{L} = \{\ell^{\lambda_i}\}_{i=0}^{6}$ of 7 candidate length-scales $\ell^{\lambda_0} = \ell^{\lambda^*}$ and $\ell^{\lambda_i} = 2(i+1)\ell^{\lambda^*}$ for $i = 1, \ldots, 6$.

The performance of our $\langle \alpha, \epsilon \rangle$-BAL policies with planning horizon length $N' = 3, 4, 5$ are compared to that of APGD and IE policies (Section 4.3.1) by running each of them on a mobile probe to direct its active sensing along a path of adjacent road segments according to the road network topology; ITE cannot be used here as it requires observing road segments separated by a pre-computed distance during exploration [Krause and Guestrin, 2007], which violates the topological constraints of the road network since the mobile probe cannot "teleport". Fig. 4.3 shows results of the tested policies averaged over 5 independent runs: It can be observed from Fig. 4.3b that our $\langle \alpha, \epsilon \rangle$-BAL policies outperform APGD and IE policies due to their nonmyopic exploration behavior.

In terms of the total online processing cost, Fig. 4.3c shows that $\langle \alpha, \epsilon \rangle$-BAL incurs < 4.5 hours given a budget of $N = 240$ road segments, which can be afforded by modern computing power. To illustrate the behavior of each policy, Figs. 4.3d-f show, respectively, the road segments observed (shaded in black) by the mobile probe running APGD, IE, and $\langle \alpha, \epsilon \rangle$-BAL policies with $N' = 5$ given a budget of $N = 60$. It can be observed from Figs. 4.3d-e that both APGD and IE cause the probe to move away from the slip roads and highways to low-speed segments whose measurements vary much more smoothly; this is expected due to their myopic exploration behavior. In contrast, $\langle \alpha, \epsilon \rangle$-BAL nonmyopically plans the probe's path and can thus

direct it to observe the more informative slip roads and highways with highly varying measurements (Fig. 4.3f) to achieve better performance.

Chapter 5

Scalable Predictive Modeling Platforms for Active Learning

This chapter introduces a novel framework of inverse variational inference to theoretically derive a non-trivial, concave objective functional (of distributions) whose optimum coincides with the predictive distribution of a chosen SGP model (Section 5.2.1). This effectively allows us to construct an alternative numerical computation of our model by iteratively following the stochastic gradient of the objective function. The proposed framework is then able to process massive datasets containing millions of data points on a single-core machine. More interestingly, we show that the complexity of each iteration can be made independent of the size of the dataset if the covariance structure of the given model satisfies certain decomposability conditions (Section 5.2.2). Examples of such SGP models include those described in [Quiñonero-Candela and Rasmussen, 2005] and [Snelson, 2007] which profess similar conditional independence structures. Empirically, we demonstrate the competitive performance of our proposed framework on a variety of real-world datasets; one of which contains more than 2 millions data points (Section 5.3).

5.1 Background and Notations

This section briefly summarizes relevant backgrounds of SGP approximations and variational inference in GP context to introduce notations and derive expressions which are necessary to understand our main results.

5.1.1 Exact GP Inference

Specifically, let $\mathbf{D} = \{\mathbf{x}_i, y_i\}_{i=1}^n$ denote our dataset which consists of n pairs of vector input $\mathbf{x}_i$ and the corresponding noisy observation y_i of its latent output $f(\mathbf{x}_i)$. The regression problem is then formulated as follows: Given $\mathbf{D}$ and an arbitrary input $\mathbf{x}_*$, we want to predict its latent output $f(\mathbf{x}_*)$. GP addresses this problem by assuming that for any set of inputs $\mathbf{X} = \{\mathbf{x}_i\}_{i=1}^n \subseteq \boldsymbol{\mathcal{X}}^n$, the random vector composing of their latent outputs $\mathbf{f}_n \triangleq [f(\mathbf{x}_1) \dots f(\mathbf{x}_n)]^T$ is distributed by a Gaussian distribution $p(\mathbf{f}_n) \triangleq \mathcal{N}(\mathbf{f}_n|\mathbf{0}, \mathbf{K}_{nn})$; its covariance matrix $\mathbf{K}_{nn} \triangleq [k(\mathbf{x}_i, \mathbf{x}_j)]_{ij}$ is commonly specified by using an anisotropic kernel function [Rasmussen and Williams, 2006]

$$k(\mathbf{x}_i, \mathbf{x}_j) \;\; \triangleq \;\; \sigma_s^2 \exp\left(-\frac{1}{2}\left(\mathbf{x}_i - \mathbf{x}_j\right)^T \boldsymbol{\Lambda}^{-1}\left(\mathbf{x}_i - \mathbf{x}_j\right)\right) \tag{5.1}$$

with $\boldsymbol{\Lambda} \triangleq \text{diag}\left[\ell_1^2, \dots, \ell_p^2\right]$ and σ_s^2 being its defining parameters. In addition, we further assume that given any set of latent outputs $\mathbf{f}_n$, the corresponding noisy observations $\mathbf{y}_n \triangleq [y_1 \dots y_n]^T$ are also distributed by a Gaussian distribution $p(\mathbf{y}_n|\mathbf{f}_n) \triangleq \mathcal{N}(\mathbf{y}_n|\mathbf{f}_n, \sigma_n^2\mathbf{I})$ where σ_n^2 denotes our noise parameter. The predictive distribution of $f_* \triangleq f(\mathbf{x}_*)$, $p(f_*|\mathbf{y}_n)$, can then be analytically evaluated in closed-form:

$$p(f_*|\mathbf{y}_n) \;\; = \;\; \mathcal{N}\left(f_* \mid \mathbf{K}_{*n}(\mathbf{K}_{nn} + \sigma_n^2\mathbf{I})^{-1}\mathbf{y}_n \;,\; \mathbf{K}_{**} - \mathbf{K}_{*n}(\mathbf{K}_{nn} + \sigma_n^2\mathbf{I})^{-1}\mathbf{K}_{n*}\right) \;,$$

where $\mathbf{K}_{*n} \triangleq [k(\mathbf{x}_*, \mathbf{x}_i)]_i$ and $\mathbf{K}_{n*} \triangleq \mathbf{K}_{*n}^T$. This, however, incurs $\mathcal{O}(n^3)$ processing time [Rasmussen and Williams, 2006] and hence, limits the use of exact GP inference to less than a few thousands data points.

5.1.2 Sparse GP Review

To reduce the prohibitively expensive cost of exact GP inference, SGPs approximate $p(f_*|\mathbf{y}_n)$ using a small set of m *inducing* variables $\mathbf{f}_m \triangleq [f(\mathbf{u}_1), \ldots, f(\mathbf{u}_m)]^T$ which are drawn from the same GP[1] and correspond to an additional set of *inducing* inputs $\mathbf{U} = \{\mathbf{u}_i\}_{i=1}^m$. The term *inducing* originates from the fundamental assumption of SGPs [Snelson and Ghahramani, 2007] that given $\mathbf{f}_m$, the conditional distribution of $(f_*, \mathbf{f}_n)$ factorizes across a pre-defined partition $\boldsymbol{\mathcal{X}} \triangleq \cup_{i=1}^p \mathbf{B}_i$ $(\mathbf{B}_i \cap \mathbf{B}_j = \emptyset)$ of the input space $\boldsymbol{\mathcal{X}}$. Thus, suppose $\mathbf{x}_* \in \mathbf{B}_p$, it follows that

$$p\left(f_*, \mathbf{f}_n|\mathbf{f}_m\right) \;=\; p\left(f_*, \mathbf{f}_p|\mathbf{f}_m\right) \left(\prod_{i=1}^{p-1} p\left(\mathbf{f}_i|\mathbf{f}_m\right)\right) \;, \tag{5.2}$$

where $\mathbf{f}_i \triangleq [f(\mathbf{x})]_{\mathbf{x}\in\mathbf{B}_i\cap\mathbf{X}}^T$ denotes the vector of latent outputs associated with training inputs in partition $\mathbf{B}_i$. Exploiting this conditional factorization (5.2), we can then derive (see Remark 1) a general framework to approximate $p(f_*|\mathbf{y}_n)$ (as detailed in Eq. (5.4) below), which is capable of interpreting the existing class of *low-rank covariance approximation*[2] SGPs directly [Titsias, 2009] or indirectly [Quiñonero-Candela and Rasmussen, 2005; Snelson and Ghahramani, 2007] (see Remark 2):

$$p(f_*|\mathbf{y}_n) \;=\; \int p\left(f_*|\mathbf{y}_p, \mathbf{f}_m\right) p\left(\mathbf{f}_m|\mathbf{y}_n\right) \mathrm{d}\mathbf{f}_m \tag{5.3}$$

$$\simeq \; \int q^*\left(f_*|\mathbf{y}_p, \mathbf{f}_m\right) q^*\left(\mathbf{f}_m\right) \mathrm{d}\mathbf{f}_m \;, \tag{5.4}$$

[1] $p(\mathbf{f}_m) \triangleq \mathcal{N}(\mathbf{f}_m|\mathbf{0}, \mathbf{K}_{mm})$ with $\mathbf{K}_{mm} \triangleq [k(\mathbf{u}_i, \mathbf{u}_j)]_{ij}$

[2] The term *low-rank covariance approximation* generally means the exact covariance is approximated by a lower-rank matrix that helps to evaluate Eq. (5.4) efficiently.

where $\mathbf{y}_p \subseteq \mathbf{y}_n$ denotes our noisy observation of $\mathbf{f}_p$ while $q^*(f_*|\mathbf{y}_p, \mathbf{f}_m)$ and $q^*(\mathbf{f}_m)$ specify the low-rank covariance approximations of $p\left(f_*|\mathbf{y}_p, \mathbf{f}_m\right)$ and $p(\mathbf{f}_m|\mathbf{y}_n)$. In practice, $q^*(f_*|\mathbf{y}_p, \mathbf{f}_m)$ is usually set as the exact conditionals $p(f_*|\mathbf{y}_p, \mathbf{f}_m)$ or $p(f_*|\mathbf{f}_m)$ whose evaluation and storage complexities are independent of n (Appendix E.4.4). As such, the majority of research pertaining to this class of SGPs has primarily focused on approximating $p(\mathbf{f}_m|\mathbf{y}_n)$ directly [Titsias, 2009] or indirectly via modifying $p(\mathbf{f}_n|\mathbf{f}_m)$ [Quiñonero-Candela and Rasmussen, 2005]. This generally results in an efficient suite of low-rank covariance approximations $q^*(\mathbf{f}_m) \simeq p(\mathbf{f}_m|\mathbf{y}_n)$ that allow (5.4) to be evaluated analytically in $\mathcal{O}(nm^2)$ (Appendix E.4.1).

Remark 1. Note that Eq. (5.3) above is the exact expression of $p(f_*|\mathbf{y}_n)$ (see its derivation in Appendix E.3) which is then approximated by replacing $p(\mathbf{f}_m|\mathbf{y}_n)$ and $p(f_*|\mathbf{y}_p, \mathbf{f}_m)$ with $q^*(\mathbf{f}_m)$ and $q^*(f_*|\mathbf{y}_p, \mathbf{f}_m)$, respectively. In addition, if f_* and $\mathbf{f}_n$ are conditionally independent given $\mathbf{f}_m$ [Quiñonero-Candela and Rasmussen, 2005], Eqs. (5.3) and (5.4) are further simplified by replacing $p(f_*|\mathbf{y}_p, \mathbf{f}_m)$ and $q^*(f_*|\mathbf{y}_p, \mathbf{f}_m)$ with $p(f_*|\mathbf{f}_m)$ and $q^*(f_*|\mathbf{f}_m)$, respectively.

Remark 2. Eq. (5.4) directly generalizes the approximated equation introduced in [Titsias, 2009], which can be straight-forwardly recovered by setting $q^*(f_*|\mathbf{y}_p, \mathbf{f}_m) = p(f_*|\mathbf{f}_m)$. Interestingly, it is also possible to set $q^*(\mathbf{f}_m)$ and $q^*(f_*|\mathbf{y}_p, \mathbf{f}_m)$ so that the resulting predictive distribution in Eq. (5.4) coincides with those of Quiñonero-Candela and Rasmussen [2005] and Snelson and Ghahramani [2007], thus inducing their GP low-rank approximation frameworks (Appendix E.4.1).

Lastly, we wrap up our SGP review here with a brief note on the motivation of this paper which distinguishes our work from the existing literature:

Motivation. Instead of diverting our effort to structure a new approximation $q^*(\mathbf{f}_m)$, we investigate a class of *numerical approaches* which *asymptotically construct* the existing $q^*(\mathbf{f}_m)$ without having to evaluate them directly in $\mathcal{O}(nm^2)$. This helps us to avoid incurring a factor of n in the processing cost which is becoming a computational bottleneck in this era of big data, thus producing a more powerful suite of *anytime* approximated SGP models (Section 5.2.2).

5.1.3 Variational Inference for Sparse GP

As the evaluation and storage complexities of $q^*(f_*|\mathbf{y}_p, \mathbf{f}_m)$ is independent of n (Section 5.1.2), the computational efficiency of the induced predictive distribution (5.4) entirely depends on how $q^*(\mathbf{f}_m) \simeq p(\mathbf{f}_m|\mathbf{y}_n)$ is constructed. In fact, this is conducted separately with the formulation in Section 5.1.2 except that the resulting $q^*(\mathbf{f}_m)$ is plugged into (5.4) to derive $p(f_*|\mathbf{y}_n)$. This section reviews a principled method to achieve this using variational inference [Titsias, 2009].

Let us begin by first introducing the fundamental idea of variational inference: An approximation to the posterior distribution of latent variables (e.g., $p(\mathbf{f}_n, \mathbf{f}_m|\mathbf{y}_n)$) is derived analytically by minimizing their KL distance, assuming it factorizes in particular ways or has specific parametric forms which are inexpensive to evaluate [Bishop, 2006]. In the GP context, Titsias [2009] parameterizes the posterior approximation $q(\mathbf{f}_n, \mathbf{f}_m) \simeq p(\mathbf{f}_n, \mathbf{f}_m|\mathbf{y}_n)$ as

$$q(\mathbf{f}_n, \mathbf{f}_m) \triangleq p(\mathbf{f}_n|\mathbf{f}_m)\, q(\mathbf{f}_m)\ , \tag{5.5}$$

where $p(\mathbf{f}_n|\mathbf{f}_m)$ is the exact GP conditional [Rasmussen and Williams, 2006] and $q(\mathbf{f}_m) \triangleq \mathcal{N}(\mathbf{f}_m|\boldsymbol{\mu}_+, \boldsymbol{\Sigma}_+)$. This naturally raises the question of how do we specify $\boldsymbol{\mu}_+$ and $\boldsymbol{\Sigma}_+$ as functions of the training data $\mathbf{D} = \{\mathbf{x}_i, y_i\}_{i=1}^n$ to minimize the KL

distance $\mathrm{KL}(q(\mathbf{f}_n, \mathbf{f}_m) \| p(\mathbf{f}_n, \mathbf{f}_m | \mathbf{y}_n))$. To address this question, we put forward the following result:

Lemma 6. *For any density function $q(\mathbf{f}_n, \mathbf{f}_m)$ and an arbitrary joint distribution $p(\mathbf{f}_n, \mathbf{f}_m, \mathbf{y}_n)$, the corresponding log marginal $\log p(\mathbf{y}_n)$ can be decomposed into functionals of $q(\mathbf{f}_n, \mathbf{f}_m)$ as detailed below:*

$$\log p(\mathbf{y}_n) \;\; = \;\; \mathcal{L}(q) + \mathrm{KL}\left(q(\mathbf{f}_n, \mathbf{f}_m) \parallel p(\mathbf{f}_n, \mathbf{f}_m | \mathbf{y}_n)\right) \;\; , \tag{5.6}$$

where we define the auxiliary functional $\mathcal{L}(q)$ as

$$\mathcal{L}(q) \;\; \triangleq \;\; \int q(\mathbf{f}_n, \mathbf{f}_m) \log \frac{p(\mathbf{y}_n, \mathbf{f}_n, \mathbf{f}_m)}{q(\mathbf{f}_n, \mathbf{f}_m)} \mathrm{d}\mathbf{f}_n \mathrm{d}\mathbf{f}_m \;\; . \tag{5.7}$$

Proof. See Appendix E.1 for a detailed proof. □

Lemma 6 thus implies minimizing $\mathrm{KL}(q(\mathbf{f}_n, \mathbf{f}_m) \| p(\mathbf{f}_n, \mathbf{f}_m | \mathbf{y}_n))$ is equivalent to maximizing $\mathcal{L}(q)$ since $p(\mathbf{y}_n)$ is constant with respect to $q(\mathbf{f}_n, \mathbf{f}_m)$. Furthermore, using the parameterization in (5.5), the *functional* $\mathcal{L}(q)$ can be cast as a *concave function* of $\boldsymbol{\mu}_+$ and $\boldsymbol{\Sigma}_+$ which maximizes when its gradient equals zero. As a result, $q(\mathbf{f}_m) \triangleq \mathcal{N}(\mathbf{f}_m | \boldsymbol{\mu}_+, \boldsymbol{\Sigma}_+)$ can be optimized by solving for $\boldsymbol{\mu}_+$ and $\boldsymbol{\Sigma}_+$ such that $\partial \mathcal{L} / \partial \boldsymbol{\mu}_+ = 0$ and $\partial \mathcal{L} / \partial \boldsymbol{\Sigma}_+ = 0$.

Remark 1. Note that Lemma 6 generally applies to any $p(\mathbf{f}_n, \mathbf{f}_m, \mathbf{y}_n)$ which includes the induced joint distribution over $(\mathbf{f}_n, \mathbf{f}_m, \mathbf{y}_n)$ of GP. Then, as $\mathrm{KL}(.\|.)$ is always non-negative, it follows that $\log p(\mathbf{y}_n) \geq \mathcal{L}(q)$ which recovers the GP variational lower-bound of [Titsias, 2009][3] if we set $p(\mathbf{f}_n, \mathbf{f}_m, \mathbf{y}_n)$ as the exact GP joint distribution.

[3] Titsias [2009] uses Jensen inequality to prove this result directly without using Lemma 6.

Remark 2. The work of Titsias [2009] is originally intended to jointly optimize $q(\mathbf{f}_m)$, the pseudo/inducing inputs $\mathbf{U} = \{\mathbf{u}_i\}_{i=1}^m$ as well as the hyper-parameters of the covariance function $k(.,.)$ (Section 5.1.1). In the context of our work here, assuming the inducing inputs and the hyper-parameters are given, the optimal $q(\mathbf{f}_m) \equiv q^*(\mathbf{f}_m)$ induces the exact predictive distribution of DTC [Seeger *et al.*, 2003] if $q^*(f_*|\mathbf{y}_p, \mathbf{f}_m) = p(f_*|\mathbf{f}_m)$ [Titsias, 2009].

5.2 Inverse Variational Inference

This section introduces a novel, interesting use of variational inference, which we term *inverse variational inference*, to theoretically construct a concave functional $\mathcal{L}(q)$ whose maximum coincides with a given distribution $q^*(\mathbf{f}_m)$ of our choice. The resulting functional then reveals an iterative procedure to evaluate $q^*(\mathbf{f}_m)$ numerically by initializing an arbitrary estimation and gradually improving it by taking small steps in the direction of the stochastic gradient of $\mathcal{L}(q)$. This iterative procedure can, in fact, be guaranteed to *asymptotically converge* towards q^* if we schedule the step sizes appropriately [Robbins and Monro, 1951]. In practice, this approach is particularly useful if the evaluation of the stochastic gradient of $\mathcal{L}(q)$ is computationally efficient (i.e., not incurring a factor of n in its complexity) as it will provide a formal trade-off between the computing expense and the estimation accuracy of $q^*(\mathbf{f}_m)$. In general, this idea is suitable for any SGP model $q^*(\mathbf{f}_m)$ satisfies the following requirements:

C1. *There exists a concave, differentiable functional $\mathcal{L}(q)$ which attains its maximum value at $q(\mathbf{f}_m) \equiv q^*(\mathbf{f}_m)$.*

C2. *The evaluation of its stochastic gradient does not incur a factor of n (i.e., the size of the dataset) in its complexity.*

In the remaining of this section, we show that for any valid choice of $q^*(\mathbf{f}_m)$, one can always construct a concave functional $\mathcal{L}(q)$ that satisfies **C1** with $q^*(\mathbf{f}_m)$ using our inverse variational inference framework (Section 5.2.1). Then, in Section 5.2.2, we further establish sufficient conditions for $q^*(\mathbf{f}_m)$ to satisfy **C2** which interestingly creates a powerful suite of scaled-up SGPs to deal with big data.

Remark. While there might exist other trivial functionals $\mathcal{L}(q)$ which is maximized at $q^*(\mathbf{f}_m) \triangleq \mathcal{N}(\mathbf{f}_m|\boldsymbol{\mu}_m, \boldsymbol{\Sigma}_m)$ (**C1**), it is unclear whether one can establish sufficient conditions for $q^*(\mathbf{f}_m)$ to satisfy **C2** with an arbitrary $\mathcal{L}(q)$. For example, one can trivially parameterize $q(\mathbf{f}_m) \triangleq \mathcal{N}(\mathbf{f}_m|\boldsymbol{\mu}_+, \boldsymbol{\Sigma}_+)$ and consequently set

$$\mathcal{L}(q) \triangleq -\frac{1}{2}(\boldsymbol{\mu}_+ - \boldsymbol{\mu}_m)^T\mathbf{A}_1(\boldsymbol{\mu}_+ - \boldsymbol{\mu}_m) - \frac{1}{2}(\boldsymbol{\Sigma}_+ - \boldsymbol{\Sigma}_m)^T\mathbf{A}_2(\boldsymbol{\Sigma}_+ - \boldsymbol{\Sigma}_m)$$

to meet **C1** although it is not even trivial to derive its *stochastic* gradient with respect to $\boldsymbol{\mu}_+$ and $\boldsymbol{\Sigma}_+$, let alone guaranteeing that its computational efficiency meets **C2**, if $\mathbf{A}_1$ and $\mathbf{A}_2$ are given arbitrarily. This motivates the use of our inverse variational inference here which is well-established to meet both **C1** and **C2**.

5.2.1 Constructing $\mathcal{L}(q)$

Specifically, we assume $q(\mathbf{f}_n, \mathbf{f}_m)$ follows the factorization in (5.5) and that $q^*(\mathbf{f}_m) = \mathcal{N}(\mathbf{f}_m|\boldsymbol{\mu}_m(\mathbf{D}), \boldsymbol{\Sigma}_m(\mathbf{D}))$ is given with $\boldsymbol{\mu}_m(\mathbf{D})$ and $\boldsymbol{\Sigma}_m(\mathbf{D})$ being represented as functions[4] of the data $\mathbf{D}$. To avoid notation cluttering, we refer to them as $\boldsymbol{\mu}_m$ and $\boldsymbol{\Sigma}_m$ hereafter but the readers should keep in mind that they are treated as functions of the data rather than some constants. Our goal here is to derive $p(\mathbf{f}_n, \mathbf{f}_m, \mathbf{y}_n)$ such

[4]For most SGPs, evaluating these functions directly incurs $\mathcal{O}(nm^2)$ processing cost (Appendix E.4.1).

that the corresponding $\mathcal{L}(q)$ is maximized at $q(\mathbf{f}_m) \equiv q^*(\mathbf{f}_m)$. This is in general a highly non-trivial task except for the special case when $q^*(\mathbf{f}_m)$ is induced from the approximated conditional $q(\mathbf{f}_n|\mathbf{f}_m)$ of DTC (Appendix E.4.1.3). In that case, it is well-known that $q^*(\mathbf{f}_m)$ happens to maximize $\mathcal{L}(q)$ when $p(\mathbf{f}_n, \mathbf{f}_m, \mathbf{y}_n)$ coincides with the exact GP joint distribution [Titsias, 2009].

Discussion. In this regard, Hensman *et al.* [2013] have taken the first step by pointing out that the given $\mathcal{L}(q)$ satisfies both **C1** and **C2** which can consequently be exploited to derive a numerical computation process for DTC. However, this work neither extends nor discusses how to derive $\mathcal{L}(q)$ for other choices of $q^*(\mathbf{f}_m)$ and under what conditions they will satisfy **C1** and **C2**. We address both of these issues in Sections 5.2.1 and 5.2.2, respectively. More critically, their proposed approach to evaluate DTC numerically also depends heavily on its structural assumptions which appears to be a special case of our general solution paradigm (Section 5.2.2.2).

Thus, the rest of this section is organized as follow: We first establish auxiliary results to simplify (Theorem 9) and analytically evaluate (Theorem 10) $\mathcal{L}(q)$ with respect to our factorization of $p(\mathbf{f}_n, \mathbf{f}_m, \mathbf{y}_n)$ and $q(\mathbf{f}_n, \mathbf{f}_m)$ in (5.11) and (5.5). Then, we show how the defining parameters of $p(\mathbf{f}_n, \mathbf{f}_m, \mathbf{y}_n)$ can be appropriately selected so that the induced $\mathcal{L}(q)$ (Lemma 6) is maximized at an arbitrary user-specified $q^*(\mathbf{f}_m) \triangleq \mathcal{N}(\mathbf{f}_m|\boldsymbol{\mu}_m, \boldsymbol{\Sigma}_m)$ (Theorem 11).

Theorem 9. *Let $q(\mathbf{f}_n, \mathbf{f}_m) = q(\mathbf{f}_n|\mathbf{f}_m)q(\mathbf{f}_m)$ and $\mathcal{L}(q)$ as defined in Lemma 6. The following equality holds:*

$$\mathcal{L}(q) \;=\; \int q(\mathbf{f}_m)\mathcal{L}_m(q)\mathrm{d}\mathbf{f}_m - \mathrm{KL}\left(q(\mathbf{f}_m)\|p(\mathbf{f}_m)\right) \;, \tag{5.8}$$

where the auxiliary functional $\mathcal{L}_m(q)$ is defined as

$$\mathcal{L}_m(q) \triangleq \int q(\mathbf{f}_n|\mathbf{f}_m) \log \frac{p(\mathbf{y}_n, \mathbf{f}_n|\mathbf{f}_m)}{q(\mathbf{f}_n|\mathbf{f}_m)} \mathrm{d}\mathbf{f}_n \ . \tag{5.9}$$

Proof. See Appendix E.2 for a detailed proof. □

Using Theorem 9, we are now ready to evaluate $\mathcal{L}(q)$ analytically. In particular, we parameterize $q(\mathbf{f}_m) \triangleq \mathcal{N}(\mathbf{f}_m|\boldsymbol{\mu}_+, \boldsymbol{\Sigma}_+)$ and factorize $q(\mathbf{f}_n, \mathbf{f}_m)$ using (5.5) which effectively set $q(\mathbf{f}_n|\mathbf{f}_m)$ as the exact GP conditional:

$$\begin{aligned} q(\mathbf{f}_n|\mathbf{f}_m) &\triangleq p(\mathbf{f}_n|\mathbf{f}_m) \\ &= \mathcal{N}\left(\mathbf{f}_n \mid \mathbf{P}\mathbf{f}_m, \mathbf{K}_{nn} - \mathbf{Q}_{nn}\right) \ , \end{aligned} \tag{5.10}$$

where $\mathbf{P} \triangleq \mathbf{K}_{nm}\mathbf{K}_{mm}^{-1}$ and $\mathbf{Q}_{nn} \triangleq \mathbf{K}_{nm}\mathbf{K}_{mm}^{-1}\mathbf{K}_{mn}$. Using (5.10), we can thus analytically represent $\mathcal{L}_m(q)$ as a quadratic function of $\mathbf{f}_m$ (Theorem 10). Then, to derive $p(\mathbf{f}_n, \mathbf{f}_m, \mathbf{y}_n)$ so that $\mathcal{L}_m(q)$ is maximized at $q^*(\mathbf{f}_m)$, we factorize:

$$p(\mathbf{f}_n, \mathbf{f}_m, \mathbf{y}_n) \triangleq p(\mathbf{y}_n|\mathbf{f}_n)p(\mathbf{f}_n|\mathbf{f}_m)p(\mathbf{f}_m) \ , \tag{5.11}$$

where $p(\mathbf{y}_n|\mathbf{f}_m)$ and $p(\mathbf{f}_n|\mathbf{f}_m)$ denote the exact GP likelihood and conditional [Rasmussen and Williams, 2006] while the exact GP prior $p(\mathbf{f}_m)$ is replaced with $p(\mathbf{f}_m) \triangleq \mathcal{N}(\mathbf{f}_m|\boldsymbol{\mu}_*, \boldsymbol{\Lambda}_*^{-1})$ ($\boldsymbol{\Lambda}_*$ denotes the precision matrix). Then, using Theorems 9 and 10, $\mathcal{L}(q)$ can now be represented as a function of $\boldsymbol{\mu}_+, \boldsymbol{\Sigma}_+, \boldsymbol{\mu}_*$ and $\boldsymbol{\Lambda}_*$ (Theorem 11). Interestingly, the obtained function is concave in both $\boldsymbol{\mu}_+$ and $\boldsymbol{\Sigma}_+$. Hence, by differentiating $\mathcal{L}(q)$ with respect to $\boldsymbol{\mu}_+$ and $\boldsymbol{\Sigma}_+$, we will be able to identify the necessary conditions for $\boldsymbol{\mu}_*$ and $\boldsymbol{\Lambda}_*$ which eliminate the derivatives at $\boldsymbol{\mu}_+ = \boldsymbol{\mu}_m$ and $\boldsymbol{\Sigma}_+ = \boldsymbol{\Sigma}_m$, thus maximizing $\mathcal{L}(q)$ at $q(\mathbf{f}_m) \equiv q^*(\mathbf{f}_m)$ (Theorem 12).

Theorem 10. *Given any set of inducing inputs* $\mathbf{U} = \{\mathbf{u}_i\}_{i=1}^m$ *along with their latent outputs* $\mathbf{f}_m$*, the functional* $\mathcal{L}_m(q)$ *can be represented as a quadratic function of* $\mathbf{f}_m$ *if* $q(\mathbf{f}_n|\mathbf{f}_m)$ *is the exact GP conditional [Rasmussen and Williams, 2006]:*

$$\mathcal{L}_m(q) \;=\; -\frac{1}{2\sigma_n^2}\mathbf{f}_m^T\mathbf{P}^T\mathbf{P}\mathbf{f}_m + \frac{1}{\sigma_n^2}\mathbf{f}_m^T\mathbf{P}^T\mathbf{y}_n + \text{const} \tag{5.12}$$

where const *absorbs all terms which are independent of both* $\mathbf{f}_n$ *and* $\mathbf{f}_m$*.*

Proof. See Appendix D.1 for a detailed proof. □

Theorem 11. *For any set of inducing inputs* $\mathbf{U} = \{\mathbf{u}_i\}_{i=1}^m$ *along with their latent outputs* $\mathbf{f}_m$*, the functional* $\mathcal{L}_m(q)$ *can be represented as a function*[5] *of* $\boldsymbol{\mu}_+$ *and* $\boldsymbol{\Sigma}_+$ *if* $q(\mathbf{f}_m) \triangleq \mathcal{N}\left(\mathbf{f}_m|\boldsymbol{\mu}_+, \boldsymbol{\Sigma}_+\right)$ *and* $p(\mathbf{f}_m) \triangleq \mathcal{N}(\mathbf{f}_m|\boldsymbol{\mu}_*, \boldsymbol{\Lambda}_*^{-1})$ *as previously assumed:*

$$\begin{aligned}\mathcal{L}(q) \;=\; & -\frac{1}{2}\boldsymbol{\mu}_+^T\mathbf{Q}\boldsymbol{\mu}_+ - \frac{1}{2}\text{tr}\left(\mathbf{Q}\boldsymbol{\Sigma}_+\right) + \frac{1}{2}\log|\boldsymbol{\Sigma}_+| \\ & + \boldsymbol{\mu}_+^T\left(\frac{1}{\sigma_n^2}\mathbf{P}^T\mathbf{y}_n + \boldsymbol{\Lambda}_*\boldsymbol{\mu}_*\right) + \text{const}\ ,\end{aligned} \tag{5.13}$$

where $\mathbf{Q} \triangleq (1/\sigma_n^2)\mathbf{P}^T\mathbf{P} + \boldsymbol{\Lambda}_*$*.*

Proof. Eq. (5.13) follows immediately by plugging (5.12) into (5.8) which can be evaluated analytically if $q(\mathbf{f}_m)$ is Gaussian. See Appendix D.2 for a detailed proof. □

Theorem 12. *If* $\boldsymbol{\Lambda}_*$ *and* $\boldsymbol{\mu}_*$ *satisfy the following conditions:*

$$\boldsymbol{\Lambda}_*\boldsymbol{\mu}_* + \frac{1}{\sigma_n^2}\mathbf{P}^T\mathbf{y}_n \;=\; \left(\frac{1}{\sigma_n^2}\mathbf{P}^T\mathbf{P} + \boldsymbol{\Lambda}_*\right)\boldsymbol{\mu}_m\ , \tag{5.14}$$

$$\boldsymbol{\Lambda}_* \;=\; \boldsymbol{\Sigma}_m^{-1} - \frac{1}{\sigma_n^2}\mathbf{P}^T\mathbf{P}\ . \tag{5.15}$$

[5]Note that we only refer to $\mathcal{L}(q)$ as a functional when the parametric form of q is undefined. Otherwise, $\mathcal{L}(q)$ can be viewed as a function of the parameters defining q.

Then, $\mathcal{L}(q)$ attains its maximum value when $\boldsymbol{\mu}_+ = \boldsymbol{\mu}_m$ and $\boldsymbol{\Sigma}_+ = \boldsymbol{\Sigma}_m$.

Proof. Plugging (5.15) into the definition of $\mathbf{Q} \triangleq (1/\sigma_n^2)\mathbf{P}^T\mathbf{P} + \boldsymbol{\Lambda}_*$ (Theorem 11), we have $\mathbf{Q} = \boldsymbol{\Sigma}_m^{-1}$. Then, substituting $\mathbf{Q} = \boldsymbol{\Sigma}_m^{-1}$ and (5.14) into (5.13) (Theorem 11), we can rewrite $\mathcal{L}(q)$ as

$$\begin{aligned} \mathcal{L}(q) &= -\frac{1}{2}\boldsymbol{\mu}_+^T\boldsymbol{\Sigma}_m^{-1}\boldsymbol{\mu}_+ - \frac{1}{2}\mathrm{tr}\left(\boldsymbol{\Sigma}_m^{-1}\boldsymbol{\Sigma}_+\right) \\ &\quad + \frac{1}{2}\log|\boldsymbol{\Sigma}_+| + \boldsymbol{\mu}_+^T\boldsymbol{\Sigma}_m^{-1}\boldsymbol{\mu}_m + \text{const} \ . \end{aligned} \tag{5.16}$$

Differentiate both sides of (5.16) with respect to $\boldsymbol{\mu}_+$ and $\boldsymbol{\Sigma}_+$, we obtain

$$\frac{\partial\mathcal{L}}{\partial\boldsymbol{\mu}_+} = -\boldsymbol{\Sigma}_m^{-1}\boldsymbol{\mu}_+ + \boldsymbol{\Sigma}_m^{-1}\boldsymbol{\mu}_m \ , \tag{5.17}$$

$$\frac{\partial\mathcal{L}}{\partial\boldsymbol{\Sigma}_+} = -\frac{1}{2}\boldsymbol{\Sigma}_m^{-1} + \frac{1}{2}\boldsymbol{\Sigma}_+^{-1} \ . \tag{5.18}$$

Thus, setting $\partial\mathcal{L}/\partial\boldsymbol{\mu}_+ = 0$ and $\partial\mathcal{L}/\partial\boldsymbol{\Sigma}_+ = 0$, it follows that $\boldsymbol{\mu}_+ = \boldsymbol{\mu}_m$ and $\boldsymbol{\Sigma}_+ = \boldsymbol{\Sigma}_m$. In addition, since (5.16) is concave in both $\boldsymbol{\mu}_+$ and $\boldsymbol{\Sigma}_+$ and there is no cross term, it is clear that $\mathcal{L}(q)$ attains its maximum value when its gradient disappears. This concludes our proof. □

Remark. Note that (5.14) and (5.15) define the space of *feasible* pairs $(\boldsymbol{\mu}_*, \boldsymbol{\Lambda}_*)$ which guarantees that $\mathcal{L}(q)$ attains its maximum value at $\boldsymbol{\mu}_+ = \boldsymbol{\mu}_m$ and $\boldsymbol{\Sigma}_+ = \boldsymbol{\Sigma}_m$. Interestingly, it is not necessary to explicitly solve for $(\boldsymbol{\mu}_*, \boldsymbol{\Lambda}_*)$ to construct the desirable $\mathcal{L}(q)$ which is maximized at $q^*(\mathbf{f}_m)$, as demonstrated in (5.16). In fact, even if (5.14) and (5.15) are *infeasible*, we can still construct $\mathcal{L}(q)$ by forcibly plugging them in (5.13) though the resulting $\mathcal{L}(q)$ cannot be interpreted as the lower bound of $p(\mathbf{y}_n)$ which does not exist.

5.2.2 Scaling Up Sparse GP for Big Data (SGP$^+$)

Using Theorem 12, a gradient-based numerical computation process which provably converges towards $(\boldsymbol{\mu}_m, \boldsymbol{\Sigma}_m)$ can now be specified. In particular, we initialize $\boldsymbol{\mu}_+^{(0)} = \mathbf{0}_m$ and $\boldsymbol{\Sigma}_+^{(0)} = \mathbf{I}_m$ and repeat the following gradient ascent update until convergence:

$$\begin{aligned} \boldsymbol{\mu}_+^{(t+1)} &= \boldsymbol{\mu}_+^{(t)} + \rho_t \frac{\partial \mathcal{L}}{\partial \boldsymbol{\mu}_+} \left(\boldsymbol{\mu}_+^{(t)}, \boldsymbol{\Sigma}_+^{(t)} \right) , \\ \boldsymbol{\Sigma}_+^{(t+1)} &= \boldsymbol{\Sigma}_+^{(t)} + \rho_t \frac{\partial \mathcal{L}}{\partial \boldsymbol{\Sigma}_+} \left(\boldsymbol{\mu}_+^{(t)}, \boldsymbol{\Sigma}_+^{(t)} \right) . \end{aligned} \tag{5.19}$$

Here, $\partial \mathcal{L}/\partial \boldsymbol{\mu}_+(\boldsymbol{\mu}_+^{(t)}, \boldsymbol{\Sigma}_+^{(t)})$ and $\partial \mathcal{L}/\partial \boldsymbol{\Sigma}_+(\boldsymbol{\mu}_+^{(t)}, \boldsymbol{\Sigma}_+^{(t)})$ denote the evaluation of (5.17) and (5.18) at $\boldsymbol{\mu}_+ = \boldsymbol{\mu}_+^{(t)}$ and $\boldsymbol{\Sigma}_+ = \boldsymbol{\Sigma}_+^{(t)}$. The above procedure is in fact guaranteed to converge if (a) $\sum_t \rho_t = +\infty$ and (b) $\sum_t \rho_t^2 < +\infty$, which is a well-known result in optimization. For example, one possible schedule is $\rho_t = \rho_0/(1 + \tau\rho_0 t)^\kappa$ where τ, κ and ρ_0 are determined empirically.

However, this offers us no computation gain as the cost of computing the exact gradient includes the cost of evaluating $q^*(\mathbf{f}_m)$ directly, which is $\mathcal{O}(nm^2)$ (Appendix E.4.1). To sidestep this issue, we adopt the *stochastic gradient ascent* (SGA) approach of Robbins and Monro [1951] which replaces the exact gradient in (5.19) by its stochastic estimation $(\partial \widehat{\mathcal{L}}/\partial \boldsymbol{\mu}_+, \partial \widehat{\mathcal{L}}/\partial \boldsymbol{\Sigma}_+)$. The idea is to quickly construct $(\partial \widehat{\mathcal{L}}/\partial \boldsymbol{\mu}_+, \partial \widehat{\mathcal{L}}/\partial \boldsymbol{\Sigma}_+)$ by randomly sampling mini-batches of m data points whose processing cost is independent of n, thus making the computation expense per iteration independent of the size of data. If $\mathbb{E}[\partial \widehat{\mathcal{L}}/\partial \boldsymbol{\mu}_+] = \partial \mathcal{L}/\partial \boldsymbol{\mu}_+$ and $\mathbb{E}[\partial \widehat{\mathcal{L}}/\partial \boldsymbol{\Sigma}_+] = \partial \mathcal{L}/\partial \boldsymbol{\Sigma}_+$, (5.19) is also guaranteed to converge using the above schedule of $\{\rho_t\}_t$ (Section 5.2.2.1).

In addition, as the standard gradient of a function (e.g., $\mathcal{L}(q)$) only points in the direction of the steepest ascent if the space of its parameters (e.g., $\boldsymbol{\mu}_+$ and $\boldsymbol{\Sigma}_+$) is

Euclidean [Amari, 1998], the numerical update in (5.19) has tacitly defined the parameter space of $q(\mathbf{f}_m)$ using the Euclidean distance between two candidate parameters, which unfortunately appears to be a poor measure of the dissimilarity between the corresponding distributions[6] [Hoffman *et al.*, 2013]. To capture a more meaningful notion of dissimilarity, we redefine the parameter space of $q(\mathbf{f}_m)$ using the symmetrized KL distance which is a natural measure of the dissimilarity between two probability distributions [Hoffman *et al.*, 2013]. Then, we derive the *natural gradient* of $\mathcal{L}(q)$ which corresponds to its standard gradient in this redefined space [Amari, 1998]. The resulting numerical approximation is therefore termed *natural gradient ascent* (NGA) as further detailed in Section 5.2.2.2.

Discussion. As the *natural gradient* of $\mathcal{L}(q)$ (in the Euclidean space) can be equivalently considered its standard gradient in the new parameter space which implements the symmetrized KL distance [Hoffman *et al.*, 2013], we can intuitively think of NGA (Section 5.2.2.2) as another version of SGA (Section 5.2.2.1) that corresponds to a different parameter space defined with a different distance metric. Both of them thus converge towards the same optimal parameters although NGA is empirically demonstrated to converge faster than SGA [Amari, 1998] when we are trying to optimize an objective function (e.g., $\mathcal{L}(q)$) with respect to a parameterized distribution (e.g., $q(\mathbf{f}_m)$). This is expected since the symmetrized KL distance is more accurate than the Euclidean distance in measuring the dissimilarity between parameterized distributions.

5.2.2.1 Approximate SGPs using SGA

This section focuses on deriving the unbiased estimation $(\partial\widehat{\mathcal{L}}/\partial\boldsymbol{\mu}_+, \partial\widehat{\mathcal{L}}/\partial\boldsymbol{\Sigma}_+)$ of the exact gradient $(\partial\mathcal{L}/\partial\boldsymbol{\mu}_+, \partial\mathcal{L}/\partial\boldsymbol{\Sigma}_+)$ for which $\mathbb{E}[\partial\widehat{\mathcal{L}}/\partial\boldsymbol{\mu}_+] = \partial\mathcal{L}/\partial\boldsymbol{\mu}_+$ and $\mathbb{E}[\partial\widehat{\mathcal{L}}/\partial\boldsymbol{\Sigma}_+] =$

[6] Interested readers are referred to Section 2.3 of [Hoffman *et al.*, 2013] for a concrete example.

$\partial\mathcal{L}/\partial\boldsymbol{\Sigma}_+$. To achieve this, the following decomposability conditions on $\boldsymbol{\mu}_m$ and $\boldsymbol{\Sigma}_m$ are necessary to facilitate the derivation of $(\partial\widehat{\mathcal{L}}/\partial\boldsymbol{\mu}_+, \partial\widehat{\mathcal{L}}/\partial\boldsymbol{\Sigma}_+)$:

Decomposability Conditions. *There exists a disjoint partition of the data* $\mathbf{D} = \bigcup_{i=1}^{p}\mathbf{D}_i$ *where* $\mathbf{D}_i = (\mathbf{X}_i, \mathbf{y}_i)$ *with* $\mathbf{X} = \bigcup_{i=1}^{p}\mathbf{X}_i$, $\mathbf{y}_n = [\mathbf{y}_1^T \ldots \mathbf{y}_p^T]^T$ *such that:*

$$\boldsymbol{\Sigma}_m^{-1} = \sum_{i=1}^{p}\mathbf{F}(m,i) + \mathbf{F}(m) \tag{5.20}$$

$$\boldsymbol{\Sigma}_m^{-1}\boldsymbol{\mu}_m = \sum_{i=1}^{p}\mathbf{G}(m,i) + \mathbf{G}(m)\ , \tag{5.21}$$

where $\mathbf{F}(m,i)$ *and* $\mathbf{G}(m,i)$ *are arbitrary functions that depend only on* $\mathbf{U} = \{\mathbf{u}_i\}_{i=1}^{m}$ *and* $\mathbf{D}_i$. *Similarly,* $\mathbf{F}(m)$ *and* $\mathbf{G}(m)$ *only depend on* $\mathbf{U}$.

Remark 1. While (5.20) and (5.21) appear rather artificial in the view of the *moment parameterization* (i.e., $\boldsymbol{\mu}_m$ and $\boldsymbol{\Sigma}_m$) of $q^*(\mathbf{f}_m)$, they can actually be viewed as the simple additive decomposability of the *natural* parameters $\boldsymbol{\theta}_1 \triangleq \boldsymbol{\Sigma}_m^{-1}\boldsymbol{\mu}_m$ and $\boldsymbol{\theta}_2 \triangleq -(1/2)\boldsymbol{\Sigma}_m^{-1}$ which define its *canonical parameterization* (Appendix E.5).

Remark 2. For any SGP model [Quiñonero-Candela and Rasmussen, 2005; Snelson and Ghahramani, 2007] which assumes factorization across a pre-defined partition $\boldsymbol{\mathcal{X}} \triangleq \cup_{i=1}^{p}\mathbf{B}_i$ $(\mathbf{B}_i \cap \mathbf{B}_j = \emptyset)$ of the input space (5.2), the corresponding disjoint partition $\{\mathbf{D}_i\}_{i=1}^{p}$ of the data is uniquely determined by setting $\mathbf{X}_i \triangleq \mathbf{B}_i \cap \mathbf{X}$.

Remark 3. Interestingly, this canonical view also reveals a systematic approach of engineering new SGPs based on the existing SGPs which satisfy (5.20) and (5.21): Given a set of *decomposable* SGP models $\{q^i(\mathbf{f}_m)\}_{i=1}^{p}$ specified by their *canonical parameterization* $\{\boldsymbol{\theta}_1^i, \boldsymbol{\theta}_2^i\}_{i=1}^{p}$ of their low-rank covariance approximations $q(\mathbf{f}_m)$, as-

suming they share the same approximated conditional $q(f_*|\mathbf{y}_p, \mathbf{f}_m)$ in (5.4), any SGP model constructed with $\widehat{\boldsymbol{\theta}}_1 \triangleq \sum_{i=1}^{p} \alpha_i \boldsymbol{\theta}_1^i$ and $\widehat{\boldsymbol{\theta}}_2 \triangleq \sum_{i=1}^{p} \alpha_i \boldsymbol{\theta}_2^i$, where $\{\alpha_i\}_{i=1}^{p}$ is the set of linear coefficients, will also satisfy (5.20) and (5.21).

In practice, the above conditions are in fact satisfied by many of the SGP models which assume the conditional independence between local latent variables given the global variables $\mathbf{f}_m$ such as SoR, DTC, FITC and PITC [Quiñonero-Candela and Rasmussen, 2005] as well as FIC and PIC [Snelson and Ghahramani, 2007]. For interested readers, the corresponding decompositions $\mathbf{F}(m), \mathbf{G}(m), \{\mathbf{F}(m,i)\}_{i=1}^{p}$ and $\{\mathbf{G}(m,i)\}_{i=1}^{m}$ of the above SGPs are derived in Appendix E.4.2. Now, suppose that the above conditions are satisfied by our choice of $\boldsymbol{\mu}_m$ and $\boldsymbol{\Sigma}_m$, the unbiased estimation $(\partial\widehat{\mathcal{L}}/\partial\boldsymbol{\mu}_+, \partial\widehat{\mathcal{L}}/\partial\boldsymbol{\Sigma}_+)$ of the exact gradient is then established in the following theorem:

Theorem 13. *Let $\mathcal{S} = \{i_l\}_{l=1}^{r}$ be a set of r i.i.d samples ($r > 0$) which are drawn from the uniform distribution over $\{1, 2, \ldots, p\}$. Then, suppose $\boldsymbol{\mu}_m$ and $\boldsymbol{\Sigma}_m$ satisfy* (5.20) *and* (5.21)*, the following stochastic estimation of the exact gradient is unbiased:*

$$\frac{\partial\widehat{L}}{\partial\boldsymbol{\mu}_+} \triangleq \mathbf{G}(m) - \mathbf{F}(m)\boldsymbol{\mu}_+ + \frac{p}{r}\sum_{l=1}^{r}\left(\mathbf{G}(m,i_l) - \mathbf{F}(m,i_l)\boldsymbol{\mu}_+\right), \tag{5.22}$$

$$\frac{\partial\widehat{L}}{\partial\boldsymbol{\Sigma}_+} \triangleq \frac{1}{2}\boldsymbol{\Sigma}_+^{-1} - \frac{1}{2}\mathbf{F}(m) - \frac{p}{2r}\sum_{l=1}^{r}\mathbf{F}(m,i_l)\,. \tag{5.23}$$

In other words, we have $\mathbb{E}_{\mathcal{S}}\left[\partial\widehat{\mathcal{L}}/\partial\boldsymbol{\mu}_+\right] = \partial\mathcal{L}/\partial\boldsymbol{\mu}_+$ and $\mathbb{E}_{\mathcal{S}}\left[\partial\widehat{\mathcal{L}}/\partial\boldsymbol{\Sigma}_+\right] = \partial\mathcal{L}/\partial\boldsymbol{\Sigma}_+$.

Proof. See Appendix D.3. □

Since (5.22) and (5.23) do not depend on n, their evaluation complexity is independent of the size of data (Appendix E.4.3). Thus, if we choose r such that $r = \mathcal{O}(m)$ then

the processing cost of evaluating $(\partial\widehat{\mathcal{L}}/\partial\boldsymbol{\mu}_+, \partial\widehat{\mathcal{L}}/\partial\boldsymbol{\Sigma}_+)$ only depends on m[7]. For SGP models such as SoR, DTC, FITC, PITC, FIC and PIC, it is easy to verify that the incurred cost of evaluating (5.22) and (5.23) is $\mathcal{O}(rm^3)$ (Appendix E.4.3). In addition, Appendix E.4.4 shows that if $q(\mathbf{f}_m)$ has already been evaluated, the cost of analytically integrating $q^*(f_*|\mathbf{f}_m, \mathbf{y}_p)$ with $q^*(\mathbf{f}_m) \equiv q(\mathbf{f}_m)$ in (5.4) (i.e., prediction cost) is independent of n. Thus, if the number of update iterations k is significantly less than n/rm, then we gain an computational advantage over traditional SGP models which often incur $\mathcal{O}(nm^2)$ processing cost (Appendix E.4.1).

5.2.2.2 Approximate SGPs using NGA

To derive the natural gradient of $\mathcal{L}(q)$, we first replace the moment parameterization of $q(\mathbf{f}_m)$ (i.e., $\boldsymbol{\mu}_+$ and $\boldsymbol{\Sigma}_+$) by its canonical counterpart $q(\mathbf{f}_m|\boldsymbol{\theta})$, as detailed below:

$$\begin{aligned} q(\mathbf{f}_m|\boldsymbol{\theta}) &= \mathcal{N}(\mathbf{f}_m|\boldsymbol{\mu}_+, \boldsymbol{\Sigma}_+) \\ &= \mathbf{h}(\mathbf{f}_m) \exp\left(\boldsymbol{\theta}^T\mathbf{T}(\mathbf{f}_m) - \mathbf{A}(\boldsymbol{\theta})\right) \end{aligned} \tag{5.24}$$

where $\mathbf{T}(\mathbf{f}_m) \triangleq \left[\mathbf{f}_m; \mathrm{vec}(\mathbf{f}_m\mathbf{f}_m^T)\right]$, $\mathbf{h}(\mathbf{f}_m) \triangleq (2\pi)^{-m/2}$ and $\mathbf{A}(\boldsymbol{\theta})$ is simply the normalizing function which guarantees that $q(\mathbf{f}_m)$ integrates to unity. Most importantly, we define the natural parameter as $\boldsymbol{\theta} \triangleq [\boldsymbol{\theta}_1; \mathrm{vec}(\boldsymbol{\theta}_2)]$ where $\boldsymbol{\theta}_1 = \boldsymbol{\Sigma}_+^{-1}\boldsymbol{\mu}_+$ and $\boldsymbol{\theta}_2 = -(1/2)\boldsymbol{\Sigma}_+^{-1}$. In particular, the metric distance which defines the parameter space is given by the Riemannian metric tensor $\mathbf{G}(\boldsymbol{\theta})$ [Amari, 1998] which corresponds to the identity matrix in case the Euclidean metric is used. Otherwise, when the parameter space implements the symmetrized KL distance, Hoffman *et al.* [2013] show that $\mathbf{G}(\boldsymbol{\theta})$ is

[7] We assume that each partition has at most m data points.

defined by the Fisher information matrix [Amari, 1998], as detailed below:

$$\mathbf{G}(\boldsymbol{\theta}) \triangleq -\mathbb{E}_{\mathbf{f}_m}\left[\frac{\partial^2 \log q(\mathbf{f}_m|\boldsymbol{\theta})}{\partial\boldsymbol{\theta}\partial\boldsymbol{\theta}^{\boldsymbol{T}}}\middle|\boldsymbol{\theta}\right] = \frac{\partial^2\mathbf{A}(\boldsymbol{\theta})}{\partial\boldsymbol{\theta}\partial\boldsymbol{\theta}^T}, \tag{5.25}$$

where the last step is formally verified in Appendix E.5.2. Then, let $\partial\mathcal{L}/\partial\boldsymbol{\theta}$ be the standard gradient of $\mathcal{L}(q)$ with respect to θ, its natural gradient $\partial\overline{\mathcal{L}}/\partial\boldsymbol{\theta}$ is defined as

$$\frac{\partial\overline{\mathcal{L}}}{\partial\boldsymbol{\theta}} \triangleq \mathbf{G}(\boldsymbol{\theta})^{-1}\frac{\partial\mathcal{L}}{\partial\boldsymbol{\theta}}. \tag{5.26}$$

To express (5.26) in terms of $\boldsymbol{\mu}_+$ and $\boldsymbol{\Sigma}_+$, let $\boldsymbol{\eta} \triangleq [\boldsymbol{\eta}_1; \mathrm{vec}(\boldsymbol{\eta}_2)]$ where $\boldsymbol{\eta}_1 = \boldsymbol{\mu}_+$ and $\boldsymbol{\eta}_2 = \boldsymbol{\mu}_+\boldsymbol{\mu}_+^T + \boldsymbol{\Sigma}_+$. We can then verify that $\mathbb{E}[\mathbf{T}(\mathbf{f}_m)] = \boldsymbol{\eta}$ (Appendix E.5.1) which implies $\partial\boldsymbol{\eta}/\partial\boldsymbol{\theta} = \mathbf{G}(\boldsymbol{\theta})$ (Appendix E.5.3). Using this result, we can rewrite (5.26) as

$$\begin{aligned}\frac{\partial\overline{\mathcal{L}}}{\partial\boldsymbol{\theta}} &\triangleq \mathbf{G}(\boldsymbol{\theta})^{-1}\frac{\partial\mathcal{L}}{\partial\boldsymbol{\theta}} \\ &= \mathbf{G}(\boldsymbol{\theta})^{-1}\frac{\partial\boldsymbol{\eta}}{\partial\boldsymbol{\theta}}\frac{\partial\mathcal{L}}{\partial\boldsymbol{\eta}} \\ &= \frac{\partial\mathcal{L}}{\partial\boldsymbol{\eta}},\end{aligned} \tag{5.27}$$

where the last step holds because $\partial\boldsymbol{\eta}/\partial\boldsymbol{\theta} = \mathbf{G}(\boldsymbol{\theta})$. Thus, we can evaluate the natural gradient in (5.26) by taking the derivative of $\mathcal{L}(q)$ with respect to $\boldsymbol{\eta}$ as in (5.27). To simplify the calculation, we take the partial derivatives of $\mathcal{L}(q)$ with respect to $\boldsymbol{\eta_1}$ and $\boldsymbol{\eta}_2$ instead of differentiating it with $\boldsymbol{\eta}$ directly. To achieve this, we first cast $\mathcal{L}(q)$ as a function of $\boldsymbol{\eta}_1$ and $\boldsymbol{\eta}_2$:

$$\begin{aligned}\mathcal{L}(q) &= -\frac{1}{2}\boldsymbol{\eta}_1^T\mathbf{Q}\boldsymbol{\eta}_1 - \frac{1}{2}\mathrm{tr}(\mathbf{Q}\boldsymbol{\eta}_2 - \mathbf{Q}\boldsymbol{\eta}_1\boldsymbol{\eta}_1^T) \\ &+ \frac{1}{2}\log|\boldsymbol{\eta}_2 - \boldsymbol{\eta}_1\boldsymbol{\eta}_1^T| + \boldsymbol{\eta}_1^T\mathbf{W} + \mathrm{const},\end{aligned} \tag{5.28}$$

with $\mathbf{Q} \triangleq (1/\sigma_n^2)\mathbf{P}^T\mathbf{P} + \mathbf{\Lambda}_*$ (Theorem 11) and $\mathbf{W} \triangleq (1/\sigma_n^2)\mathbf{P}^T\mathbf{y}_n + \mathbf{\Lambda}_*\boldsymbol{\mu}_*$. This can be straight-forwardly verified using (5.13) and $\boldsymbol{\eta}$'s definition. The natural gradient of $\mathcal{L}(q)$ is then given by

$$\frac{\partial \mathcal{L}}{\partial \boldsymbol{\eta}_1} = -\left(\boldsymbol{\eta}_2 - \boldsymbol{\eta}_1\boldsymbol{\eta}_1^T\right)^{-1}\boldsymbol{\eta}_1 + \mathbf{W} \ , \tag{5.29}$$

$$\frac{\partial \mathcal{L}}{\partial \boldsymbol{\eta}_2} = \frac{1}{2}\left(\left(\boldsymbol{\eta}_2 - \boldsymbol{\eta}_1\boldsymbol{\eta}_1^T\right)^{-1} - \mathbf{Q}\right) \ . \tag{5.30}$$

Finally, note that if we choose $(\mathbf{\Lambda}_*, \boldsymbol{\mu}_*)$ which satisfy (5.14) and (5.15) to guarantee that $\mathcal{L}(q)$ maximizes at $q(\mathbf{f}_m) \equiv q^*(\mathbf{f}_m)$ (Theorem 12), it then follows immediately that $\mathbf{Q} = \mathbf{\Sigma}_m^{-1}$ and $\mathbf{W} = \mathbf{\Sigma}_m^{-1}\boldsymbol{\mu}_m$. In addition, note that by definitions, $\boldsymbol{\theta}_1 = (\boldsymbol{\eta}_2 - \boldsymbol{\eta}_1\boldsymbol{\eta}_1^T)^{-1}\boldsymbol{\eta}_1$ and $\boldsymbol{\theta}_2 = -(1/2)(\boldsymbol{\eta}_2 - \boldsymbol{\eta}_1\boldsymbol{\eta}_1^T)^{-1}$. Hence, (5.29) and (5.30) are rewritten as

$$\frac{\partial \mathcal{L}}{\partial \boldsymbol{\eta}_1} = \mathbf{\Sigma}_m^{-1}\boldsymbol{\mu}_m - \boldsymbol{\theta}_1 \ , \tag{5.31}$$

$$\frac{\partial \mathcal{L}}{\partial \boldsymbol{\eta}_2} = -\boldsymbol{\theta}_2 - \frac{1}{2}\mathbf{\Sigma}_m^{-1} \ . \tag{5.32}$$

Eqs (5.31) and (5.32) thus reveal that if $\boldsymbol{\mu}_m$ and $\mathbf{\Sigma}_m$ satisfy the decomposability conditions (i.e., (5.20) and (5.21)) mentioned in Section 5.2.2.1, then it is also possible to derive the unbiased stochastic estimation of the exact natural gradient in (5.31) and (5.32), as formalized in the following theorem.

Theorem 14. *Let $\mathcal{S} = \{i_l\}_{l=1}^r$ be a set of r i.i.d samples ($r > 0$) which are drawn from the uniform distribution over $\{1, 2, \ldots, p\}$. Suppose $\boldsymbol{\mu}_m$ and $\mathbf{\Sigma}_m$ satisfy* (5.20) *and* (5.21)*, the following stochastic estimation of the exact natural gradient is unbiased:*

$$\frac{\partial \widehat{\mathcal{L}}}{\partial \boldsymbol{\eta}_1} \triangleq \left(\mathbf{G}(m) + \frac{p}{r}\sum_{l=1}^{r}\mathbf{G}(m, i_l)\right) - \boldsymbol{\theta}_1 \ , \tag{5.33}$$

$$\frac{\partial \widehat{\mathcal{L}}}{\partial \boldsymbol{\eta}_2} \triangleq -\boldsymbol{\theta}_2 - \frac{1}{2}\left(\mathbf{F}(m) + \frac{p}{r}\sum_{l=1}^{r}\mathbf{F}(m, i_l)\right) \ . \tag{5.34}$$

In other words, we have $\mathbb{E}_{\mathcal{S}}[\partial\widehat{\mathcal{L}}/\partial\boldsymbol{\eta}_1] = \partial\mathcal{L}/\partial\boldsymbol{\eta}_1$ *and* $\mathbb{E}_{\mathcal{S}}[\partial\widehat{\mathcal{L}}/\partial\boldsymbol{\eta}_2] = \partial\mathcal{L}/\partial\boldsymbol{\eta}_2$.

Proof. See Appendix D.4. □

Theorem 14 thus concludes our theoretical analysis which effectively establishes a powerful suite of scaled-up SGP models for big data. While the proposed theory is currently restricted to the class of SGP models which meet the decomposability conditions in (5.20) and (5.21), it appears that these conditions are in fact satisfied by the whole class of low-rank covariance approximation SGPs introduced in [Quiñonero-Candela and Rasmussen, 2005] and [Snelson, 2007] (see Appendix E.4.2). More importantly, we have shown that the conditions in (5.20) and (5.21) can also be exploited to systematically engineer new *decomposable SGPs* as *linear combinations* of the existing SGPs (see Remark 3 of Section 5.2.2.1). As a brief summary, the update equations in (5.19) can now be rewritten as

$$\boldsymbol{\theta}_1^{(t+1)} = \boldsymbol{\theta}_1^{(t)} + \rho_t \frac{\partial\widehat{\mathcal{L}}}{\partial\boldsymbol{\eta}_1}\left(\boldsymbol{\theta}_1^{(t)}, \boldsymbol{\theta}_2^{(t)}\right) , \tag{5.35}$$

$$\boldsymbol{\theta}_2^{(t+1)} = \boldsymbol{\theta}_2^{(t)} + \rho_t \frac{\partial\widehat{\mathcal{L}}}{\partial\boldsymbol{\eta}_2}\left(\boldsymbol{\theta}_1^{(t)}, \boldsymbol{\theta}_2^{(t)}\right) , \tag{5.36}$$

with $\boldsymbol{\Sigma}_+^{(t)} = -(1/2)\boldsymbol{\theta}_2^{(t)^{-1}}$ and $\boldsymbol{\mu}_+^{(t)} = -(1/2)\boldsymbol{\theta}_2^{(t)^{-1}}\boldsymbol{\theta}_1^{(t)}$. For initialization, one can start with $\boldsymbol{\theta}_1^{(0)} = \mathbf{0}_m$ and $\boldsymbol{\theta}_2^{(0)} = \mathbf{I}_m$. In particular, if $\boldsymbol{\mu}_m$ and $\boldsymbol{\Sigma}_m$ are selected as those of DTC [Seeger *et al.*, 2003; Quiñonero-Candela and Rasmussen, 2005], (5.35) and (5.36) recover the exact numerical computation of DTC [Seeger *et al.*, 2003] as proposed in [Hensman *et al.*, 2013] which thus appears to be a special case of our work here.

5.3 Experiments

This section empirically evaluates the efficiency of our proposed framework of anytime SGP+ models (Section 5.2.2) on a wide range of large-scale real-world datasets (one of which contains more than 2 millions data points):

(a) The AIMPEAK dataset [Chen *et al.*, 2013b] contains $n = 41850$ traffic observations which are collected along 775 road segments (including highways, arterials, slip roads, etc.) of an urban road network in Singapore during morning peak hours (6 - 10:30 a.m.). Each such observation records the traffic speed at a particular road segment which is represented by a 5-dimensional vector of input features including length, number of lanes, speed limit, direction and time. The traffic speeds are the outputs whose mean and standard deviation are 49.5 (km/h) and 21.7 (km/h).

(b) The SARCOS dataset [Vijayakumar *et al.*, 2005; Chen *et al.*, 2013b] contains $n = 48933$ data points pertaining to an inverse dynamic problem of a 7-degrees-of-freedom SARCOS robot arm. Each data point is a tuple of 7 joint positions, 7 joint velocities, 7 joint accelerations and 7 joint torques for which we split into (a) an input vector which comprises 21 features: 7 joint positions, 7 joint velocities, 7 joint accelerations; and (b) an output scalar which is selected as one of the 7 joint torques. The mean output is 13.7 and its standard deviation is 20.5.

(c) The UK Housing Price datasets[8] [Hensman *et al.*, 2013] of apartment ($n = 104268$) and detached house ($n = 147898$) price which contains hundreds of thousands entries

[8]The UK Housing Price dataset of apartment monthly transactions is previously used in [Hensman *et al.*, 2013] for which the authors only use a subset of 75000 data points in their experiments. For interested readers, these datasets are published at http://data.gov.uk/dataset/land-registry-monthly-price-paid-data/

of property transactions in England and Wales during 2012. Each entry archives information about the transaction price and the postal code of the property which is converted to latitude and longitude by cross-referencing against a postal code database. The input thus comprises a 2-dimensional feature vector (i.e., latitude and longitude) on which we regress the normalized logarithm of the transaction price (i.e., output).

(d) The AIRLINE dataset contains $n = 2,055,733$ records of information about every commercial flight in the USA from January to April 2008. The input is a vector comprising of 8 features: the age of the aircraft (i.e., the number of years in service), the travel distance (km), airtime, departure and arrival time (min) as well as day of the week, day of the month and month. The output is the delay time (min) of the flight featured by the corresponding input vector[9].

All datasets are modeled using GPs whose prior covariance is defined using (a) the anisotropic kernel function (Eq. (5.1)) for the AIMPEAK[10], SARCOS [Chen *et al.*, 2013b] and AIRLINE [Hensman *et al.*, 2013] datasets; (b) the sum of squared exponential covariances [Hensman *et al.*, 2013] for the UK Housing datasets

$$k(\mathbf{x},\mathbf{x}') = \sum_{j=1}^{2}\left(\left(\sigma_s^j\right)^2\exp\left(-\frac{1}{2}\sum_{i=1}^{2}\left(\frac{x_i-x_i^{'}}{\ell_i^j}\right)^2\right)\right)+\left(\sigma_s^0\right)^2+\sigma_n^2\delta_{xx'}\ , \quad (5.37)$$

which consists of (a) 2 squared exponential terms $\left\{\left(\sigma_s^j\right)^2\exp\left(-\frac{1}{2}\sum_{i=1}^{2}\left(\frac{x_i-x_i^{'}}{\ell_i^j}\right)^2\right)\right\}_{j=1}^{2}$ to account for the national and regional variations in property prices; (b) a constant variance $\left(\sigma_s^0\right)^2$ allowed for non-zero mean data; and (c) the observation noise $\sigma_n^2\delta_{xx'}$

[9]Part of this dataset has been previously used for experiments in [Hensman *et al.*, 2013] with the same input-output settings.

[10]To model this traffic dataset using GPs, the road segment features have to be embedded into the Euclidean space using multi-dimension scaling [Chen *et al.*, 2012] so that the anisotropic kernel function (5.1) can be applied.

with $\delta_{xx'} = 1$ if $x = x'$ and 0 otherwise. The hyper-parameters are learned using a randomly selected data of size 10000 via maximum likelihood estimation [Rasmussen and Williams, 2006].

For each experiment, a small subset of the entire dataset is randomly selected and set aside as test data for predictions (10% for the AIMPEAK and SARCOS datasets, 5% for the UK Housing and AIRLINE datasets). The remaining data is then partitioned into $p = k$ blocks using k-means to assume the conditional independence structure of SGPs (Section 5.1.2). Our SGP+ models which include PIC+, PITC+ and DTC+ are then evaluated[11] on all these datasets with varying k and m (Section 5.3.2). All experiments are run on a a single core of a Linux system with Intel® Xeon® E5620 at 2.4GHz with 96 GB memory and 16 cores.

5.3.1 Performance Metrics

The tested anytime SGP+ models (i.e., PIC+, PITC+ and DTC+) are evaluated using the following performance metrics:

Prediction Error. Suppose the anytime SGP+ model is given by $q(\mathbf{f}_m) = \mathcal{N}(\mathbf{f}_m|\boldsymbol{\mu}_+, \boldsymbol{\Sigma}_+)$, its induced predictive distribution $q(\mathbf{f}_*) = \mathcal{N}(\mathbf{f}_*|\mathbb{E}[\mathbf{f}_*], \mathbb{V}[\mathbf{f}_*])$ for any test input $\mathbf{x}_*$ can be analytically constructed according to Appendix E.4.4. Its prediction error is defined as the root mean square error (RMSE)

$$\text{RMSE} \;=\; \sqrt{|\boldsymbol{\mathcal{S}}_*|^{-1} \sum_{\mathbf{x}\in\boldsymbol{\mathcal{S}}_*} \left(\mathbf{y}_* - \mathbb{E}[\mathbf{f}_*]\right)^2}\;,$$

[11] We do not consider SoR, FI(T)C because (a) SoR is only different from DTC in terms of their estimation of predictive variance which will not affect their prediction's RMSE, and (b) FI(T)C are special cases of PI(T)C where we allocate one block for each input data.

which is empirically evaluated on our test set $\boldsymbol{\mathcal{S}}_*$.

Anytime Efficiency. The anytime efficiency of these anytime SGP+ models can be jointly demonstrated via the trade-off between their (a) **Time Efficiency** (TE) which increases as we reduce the number of update iterations vs. (b) **Prediction Efficiency** (PE) in comparison to those of their SGP counterparts. Formally, **Time Efficiency** is defined as the incurred time of the SGP model divided by that of its anytime SGP+ counterpart and likewise, **Performance Efficiency** is defined as the prediction error of the SGP model divided by that of its anytime SGP+ counterpart. In practice, increasing TE reduced the processing cost of SGP+ but in exchange, it degrades PE.

5.3.2 Results and Analysis

This section reports and analyzes the performance of PIC+, PITC+ and DTC+ on all datasets: (a) AIMPEAK (Figs. 5.1, 5.2, 5.3, 5.5 and 5.4), (b) SARCOS (Figs. 5.6, 5.7, 5.8, 5.9 and 5.10), (c) UK Housing Price (Figs. 5.11, 5.12, 5.13, 5.14 and 5.15) with varying support set sizes m and number k of partitions/blocks, and (d) AIRLINE with $m = 1000$ supporting points and $k = 1000$ blocks.

5.3.2.1 AIMPEAK Dataset

Figs. 5.1, 5.2 and 5.3 consistently show that the anytime predictive performance of PIC+, PITC+ and DTC+ always converge towards those of PIC, PITC and DTC with varying support set size $m = 250, 500, 750, 1000$ and number $k = 50, 75, 100$ of blocks. This verifies and supports the previously developed theory in Section 5.2.2 that the predictive distributions of the proposed anytime SGP+ models asymptotically approach those of their SGP counterparts.

Remarkably, it can also be observed from Figs. 5.1, 5.2 and 5.3 that the prediction errors of PIC+, PITC+ and DTC+ appear to decrease exponentially as the number of update iterations increases. Fig. 5.4 further investigates this in terms of the trade-off between prediction vs. time efficiency (Section 5.3.1) which essentially captures how the prediction efficiency will degrade if we boost the time efficiency of the SGP+ models to meet the real-time requirements in time-critical applications. For example, Fig. 5.14c indicates that PIC+ can preserve 80% of PIC's prediction efficiency (i.e., PE = 0.8), meaning that PIC+'s prediction error is only $1/0.8 = 1.25$ times larger than PIC's, while processing data 10 times as fast as PIC (i.e., TE = 10).

On the other hand, Fig. 5.5 reveals that PIC+ significantly outperforms PITC+ and DTC+ for all settings of m and k: Its stage-wise prediction error falls below those of PITC+ and DTC+ by a large margin. This is expected because unlike PITC [Quiñonero-Candela and Rasmussen, 2005] and DTC [Seeger *et al.*, 2003] which tacitly assume the conditional independence between the test and training outputs (i.e., f_* and $\mathbf{f}_n$) given the inducing output $\mathbf{f}_m$ (see Remark 1 after Eq. (5.4)), PIC [Snelson, 2007] does not. As a result, it is capable of exploiting both local and global information (e.g., $p(f_*|\mathbf{f}_m, \mathbf{y}_p)$ and $q(\mathbf{f}_m)$) to improve its prediction. On the contrary, PITC and DTC's conditional independence assumption technically implies $p(f_*|\mathbf{f}_m, \mathbf{y}_p) = p(f_*|\mathbf{f}_m)$ (see Remark 2 after Eq. (5.4)) and consequently, ignore the local information $\mathbf{y}_p$ which comes from the data block that contains $\mathbf{x}_*$. Then, as PIC+ (PITC+, DTC+) is designed to converge towards PIC (PITC, DTC) (Section 5.2.2), it directly inherits the above modeling advantage (disadvantage) of PIC (PITC, DTC) which empirically results in its superior predictive performance.

5.3.2.2 SARCOS and UK Housing Price Datasets

Similar to the our previous observations of the AIMPEAK dataset (Section 5.3.2.1), it can be observed from both the SARCOS and UK Housing Price datasets' empirical results that:

(a) The predictive performance of PIC+ (Figs. 5.6, 5.11a-c and 5.12a-c), PITC+ (Figs. 5.7, 5.11d-f and 5.12d-f) and DTC+ (Figs. 5.8, 5.11g-i and 5.12g-i) empirically converges towards those of PIC, PITC and DTC with varying m and k, thus providing further support for our developed theory in Section 5.2.2.

(b) PIC+ significantly outperforms both PITC+ and DTC+ on all experiment settings (Figs. 5.9 and 5.13) which is expected since PIC+ can exploit local information to improve its performance while the others cannot (see Section 5.3.2.1 for a detailed explanation). In addition, Fig. 5.13 interestingly reveals that PITC+ significantly outperforms DTC+ on the UK Housing Price datasets while maintaining a competitive performance on both the SARCOS and AIMPEAK datasets (Figs. 5.5, 5.9). Despite being a little bit more complicated and less straight-forward, this observation is in fact not unexpected considering that DTC+ inherits DTC's fundamental assumption of the deterministic relation between the support and training variables (i.e., $\mathbf{f}_m$ and $\mathbf{f}_n$) which might seriously affects its prediction when the measurement noise is low [Snelson and Ghahramani, 2007]: According to our inspection, the measurement noises of the UK Housing Price datasets are significantly lower than those of the AIMPEAK and SARCOS datasets.

(c) Figs. 5.10, 5.14 and 5.15 illustrate the anytime trade-off curve between the time vs. prediction efficiency of these SGP+ models (i.e., PIC+, PITC+ and DTC+)

which essentially explains what happens to their predictive performance if their processing time is reduced (hence, increasing the time efficiency), thus providing us with a powerful tool for making good decisions in time-critical, data-intensive applications. Besides, in most of the cases, we notice that the anytime curves of PIC+ and PITC+ appears less steep than that of DTC+ which suggests that PIC+ and PITC+ might achieve better speedup than DTC+ given the same level of predictive efficiency to maintain.

5.3.2.3 AIRLINE Dataset

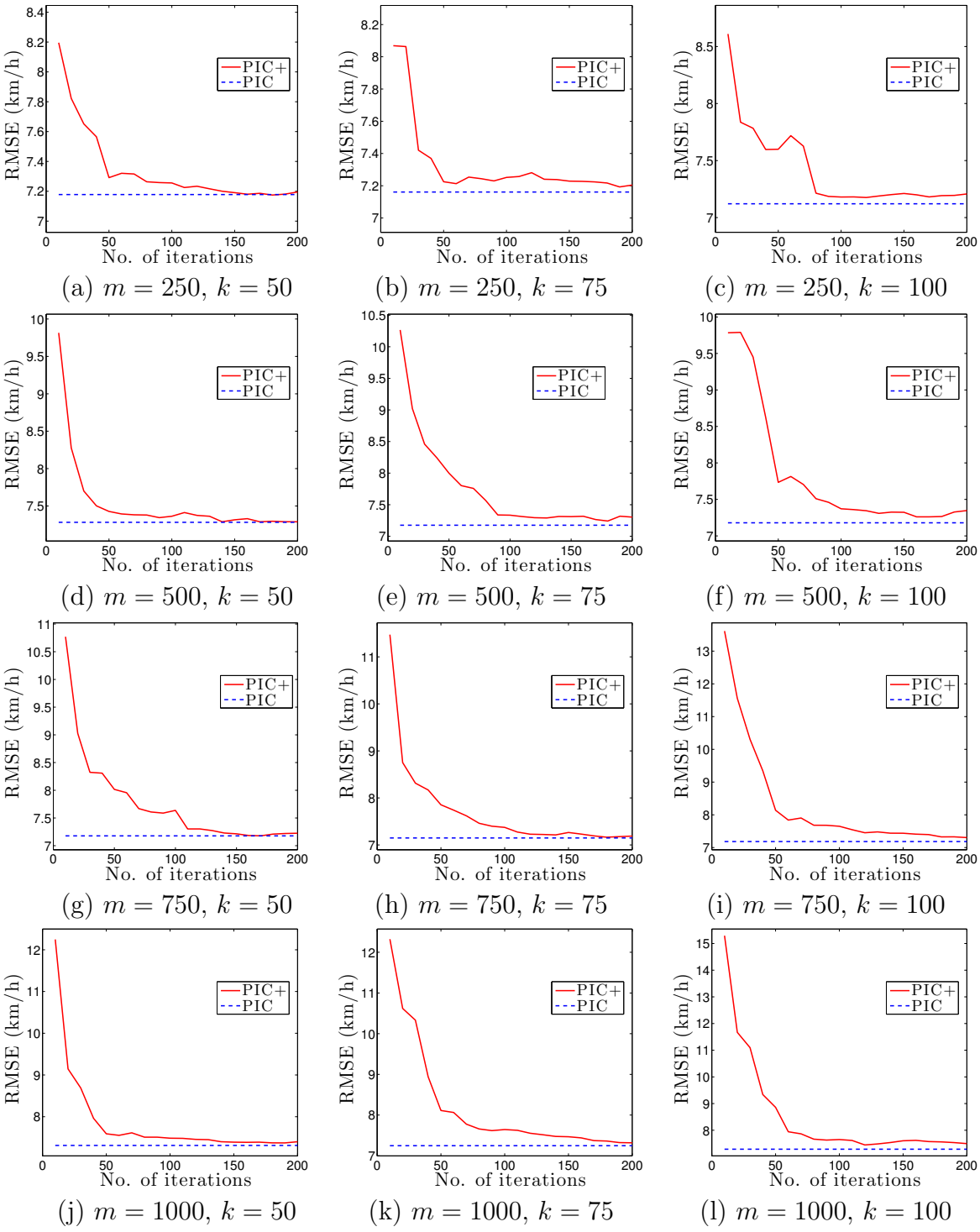
The performance of our SGP+ models (i.e., PIC+, PITC+ and DTC+) on the AIRLINE dataset are reported in Figs. 5.17 and 5.16 below. Specifically, the performance behaviors of PIC+, PITC+ and DTC+ are mostly consistent with our observations earlier with the previous datasets: (a) All SGP+ models' empirical error converges towards those of their exact SGP counterparts as we increase the number of learning iterations (Figs. 5.17a, 5.17b and 5.17c), and (b) PIC+ significantly outperforms both PITC+ and DTC+ thanks to its capability of exploiting local information to improve its prediction (Fig. 5.16a). As a matter of fact, while PIC manages to achieve an RMSE of 36.2409 (min), PITC's and DTC's are stuck at 38.6667 and 38.6818, which are about 6.73% more than PIC's.

In terms of scalability, Fig. 5.17d further indicates that PIC+ can preserve almost 100% of PIC's predictive accuracy while running 25 times as fast. It only takes PIC+ around 500 seconds to complete 60 update iterations and converge on PIC in terms of the RMSE (Figs. 5.17a, 5.17b and 5.17c) while the exact evaluation of PIC costs 8330 seconds. On the other hand, while PITC+ and DTC+ also converge quickly towards their exact SGP counterparts (i.e., PITC and DTC) at roughly the same rate, their anytime RMSE is worse than PIC+'s (Fig. 5.16a). This strongly motivates the use of

PIC+ which appears to preserve the state-of-the-art SGP performance significantly better than both PITC+ and DTC+ on large dataset. In terms of the processing cost, our empirical results indicate that PIC+, PITC+ and DTC+ incur (on average) 8.76, 8.64 and 1.42 seconds per learning iteration, respectively. Their processing time also increases linearly in the number of iterations, as shown in Fig. 5.16b.

5.3.3 Summary

Finally, to conclude our empirical analysis, this section provides a brief summary of the most important observations that we have presented and analyzed in the previous sections: (a) The predictive performance of PIC+, PITC+ and DTC+ consistently converges towards those of PIC, PITC and DTC when evaluated on all datasets with various SGP settings; (b) PIC+ significantly outperforms both PITC+ and DTC+ on all experiment settings and in addition, PITC+ also appears to significantly outperform DTC+ on the *low-noise* UK Housing Price datasets while maintaining a competitive performance on the others. This highlights the significance of having a general framework to approximate any SGP model in an anytime fashion, thus bringing more competitive learning models to the arena of big data, which is the main thrust of our work here; (c) Lastly, we also empirically analyze the anytime speedup capabilities of our PIC+, PITC+ and DTC+ models given some level of predictive efficiency to maintain via their corresponding trade-off curves, thus providing a powerful tool for making good decisions in time-critical, data-intensive applications.



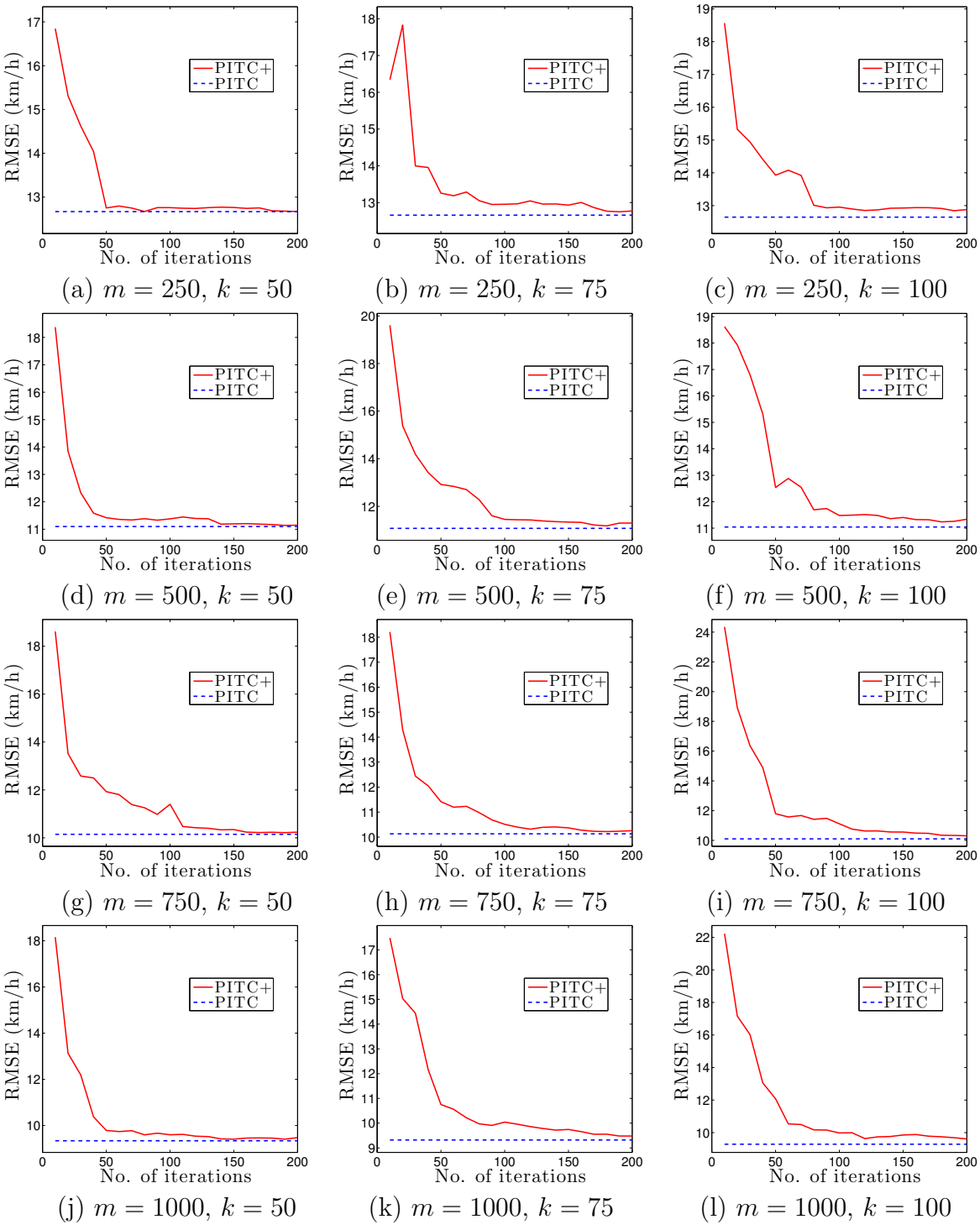
(a) $m = 250$, $k = 50$ (b) $m = 250$, $k = 75$ (c) $m = 250$, $k = 100$

(d) $m = 500$, $k = 50$ (e) $m = 500$, $k = 75$ (f) $m = 500$, $k = 100$

(g) $m = 750$, $k = 50$ (h) $m = 750$, $k = 75$ (i) $m = 750$, $k = 100$

(j) $m = 1000$, $k = 50$ (k) $m = 1000$, $k = 75$ (l) $m = 1000$, $k = 100$

Figure 5.1: PIC+'s anytime prediction error empirically converges towards that of PIC on the AIMPEAK dataset with varying support set size $m = 250, 500, 750, 1000$ and number $k = 50, 75, 100$ of blocks.



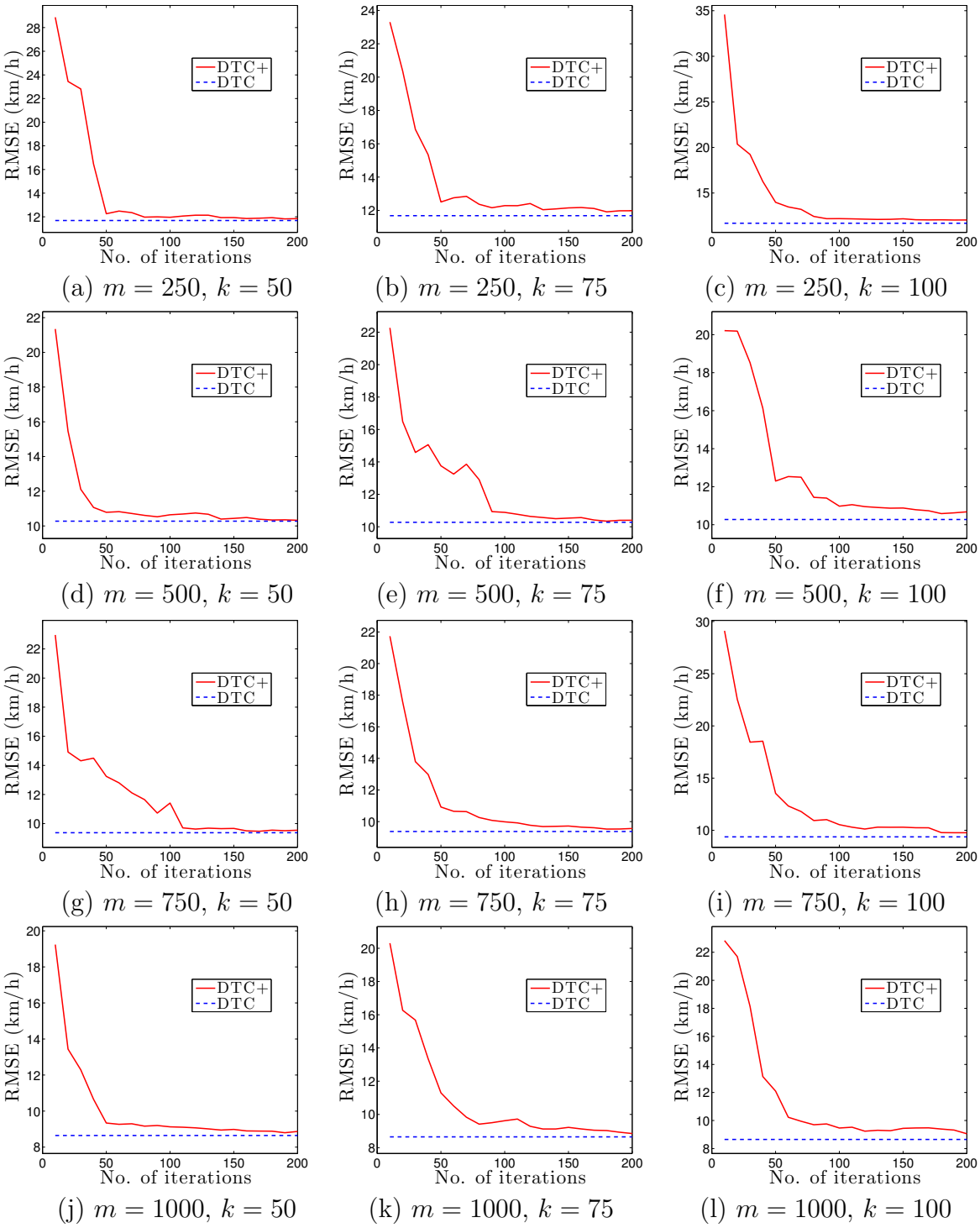
(a) $m = 250$, $k = 50$ (b) $m = 250$, $k = 75$ (c) $m = 250$, $k = 100$

(d) $m = 500$, $k = 50$ (e) $m = 500$, $k = 75$ (f) $m = 500$, $k = 100$

(g) $m = 750$, $k = 50$ (h) $m = 750$, $k = 75$ (i) $m = 750$, $k = 100$

(j) $m = 1000$, $k = 50$ (k) $m = 1000$, $k = 75$ (l) $m = 1000$, $k = 100$

Figure 5.2: PITC+'s anytime prediction error empirically converges towards that of PITC on the AIMPEAK dataset with varying support set size $m = 250, 500, 750, 1000$ and number $k = 50, 75, 100$ of blocks.



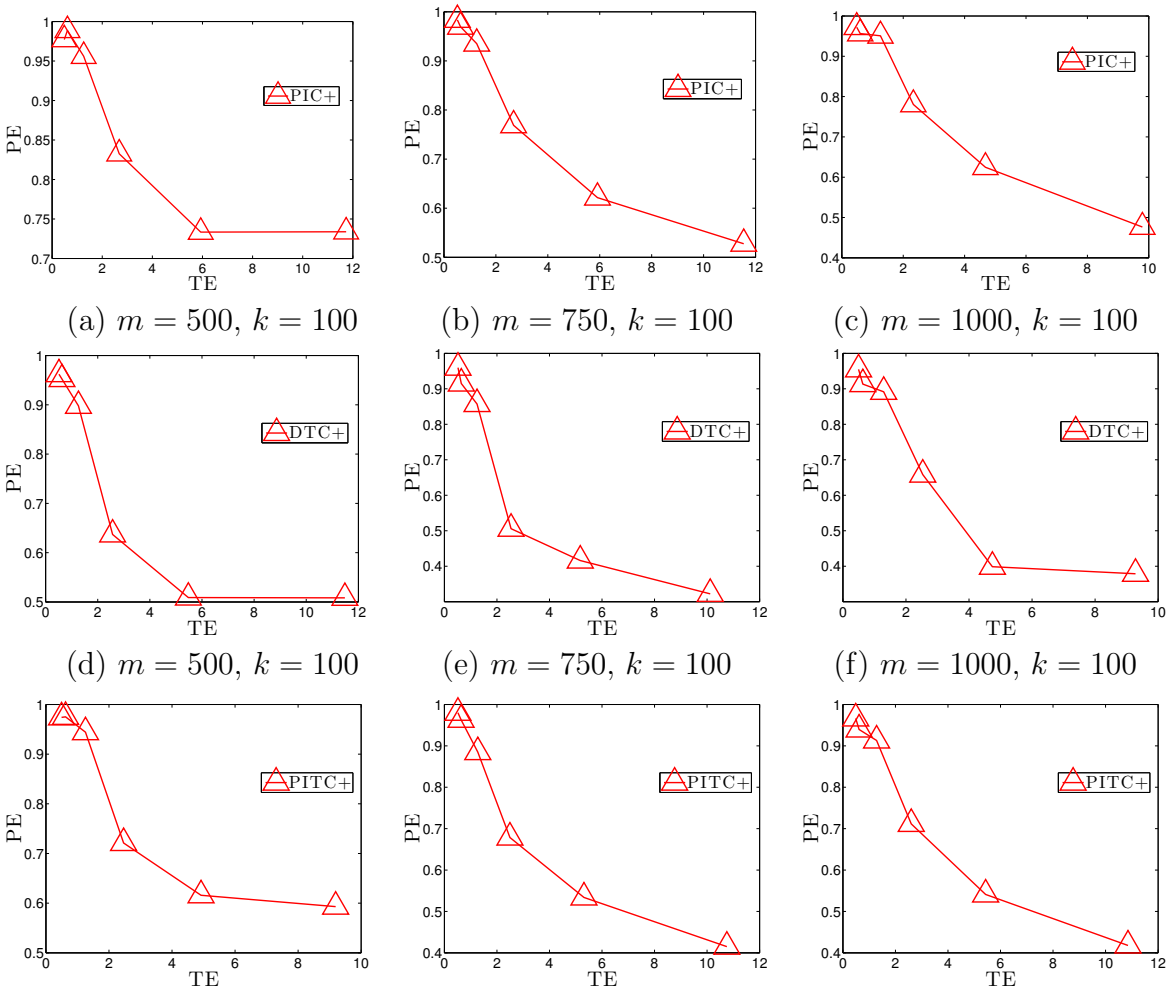
(a) $m = 250,\ k = 50$ (b) $m = 250,\ k = 75$ (c) $m = 250,\ k = 100$

(d) $m = 500,\ k = 50$ (e) $m = 500,\ k = 75$ (f) $m = 500,\ k = 100$

(g) $m = 750,\ k = 50$ (h) $m = 750,\ k = 75$ (i) $m = 750,\ k = 100$

(j) $m = 1000,\ k = 50$ (k) $m = 1000,\ k = 75$ (l) $m = 1000,\ k = 100$

Figure 5.3: DTC+'s anytime prediction error empirically converges towards those of DTC on the AIMPEAK dataset with varying support set size $m = 250, 500, 750, 1000$ and number $k = 50, 75, 100$ of blocks.

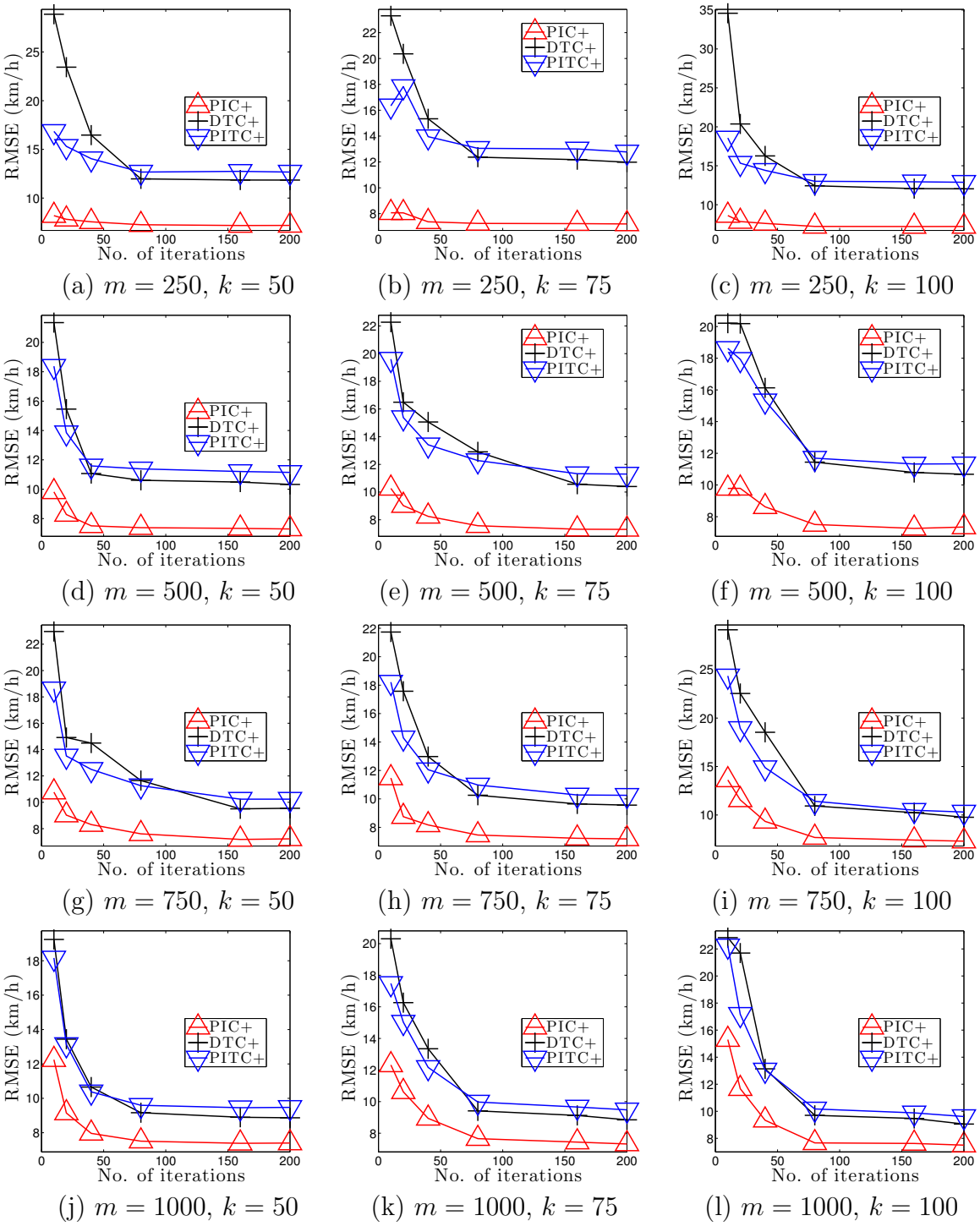


(a) $m = 500$, $k = 100$ (b) $m = 750$, $k = 100$ (c) $m = 1000$, $k = 100$

(d) $m = 500$, $k = 100$ (e) $m = 750$, $k = 100$ (f) $m = 1000$, $k = 100$

(g) $m = 500$, $k = 100$ (h) $m = 750$, $k = 100$ (i) $m = 1000$, $k = 100$

Figure 5.4: Graphs of time vs. prediction efficiency (TE vs. PE) trade-off for (a-c) PIC+, (d-f) DTC+ and (g-i) PITC+ evaluated on the AIMPEAK dataset with varying support set size $m = 250, 500, 750, 1000$ and number $k = 100$ of blocks.



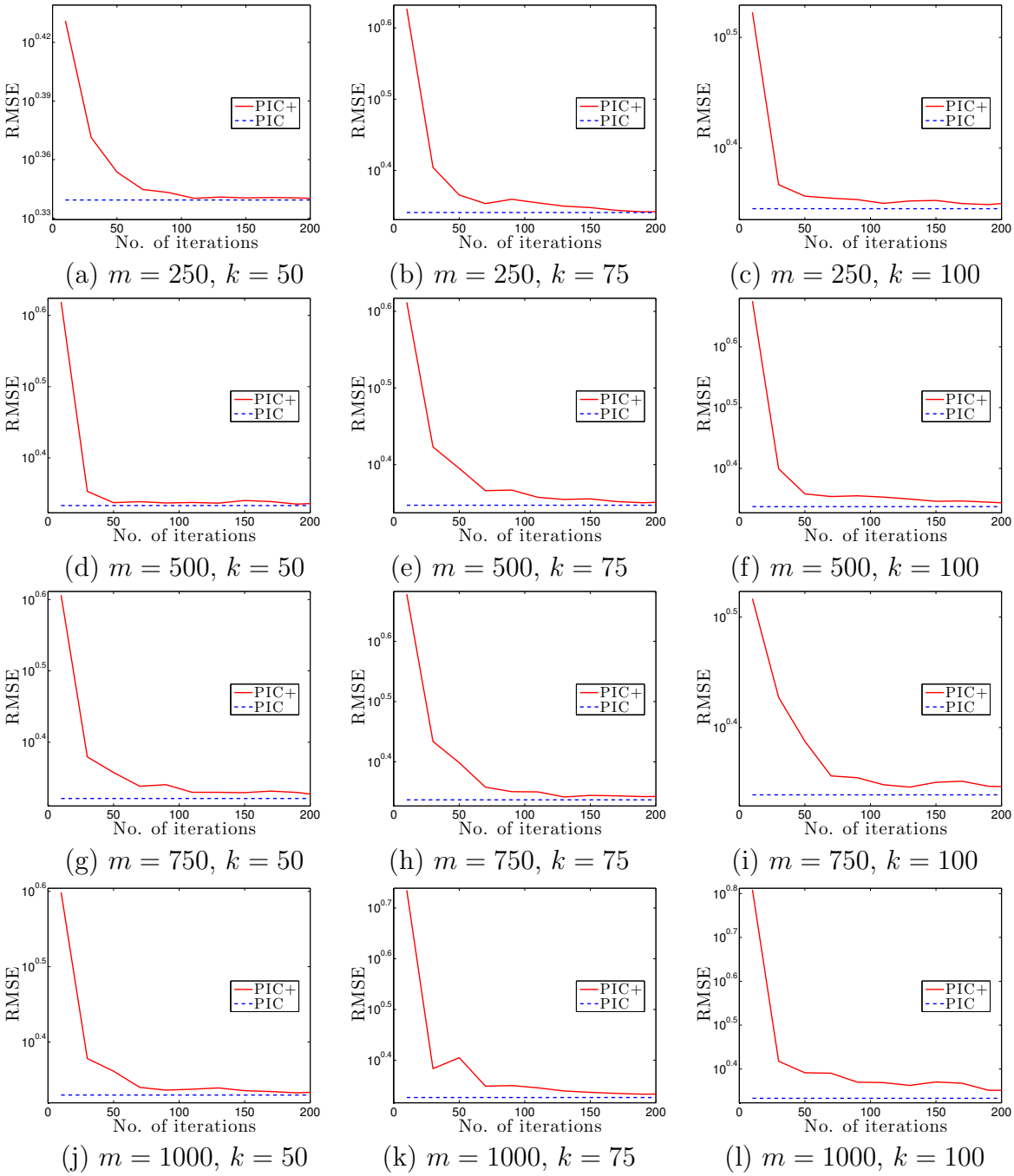
(a) $m = 250$, $k = 50$ (b) $m = 250$, $k = 75$ (c) $m = 250$, $k = 100$

(d) $m = 500$, $k = 50$ (e) $m = 500$, $k = 75$ (f) $m = 500$, $k = 100$

(g) $m = 750$, $k = 50$ (h) $m = 750$, $k = 75$ (i) $m = 750$, $k = 100$

(j) $m = 1000$, $k = 50$ (k) $m = 1000$, $k = 75$ (l) $m = 1000$, $k = 100$

Figure 5.5: Graphs of the anytime RMSE of PIC+, PITC+ and DTC+ evaluated on the AIMPEAK dataset with varying support set size $m = 250, 500, 750, 1000$ and number $k = 50, 75, 100$ of blocks.



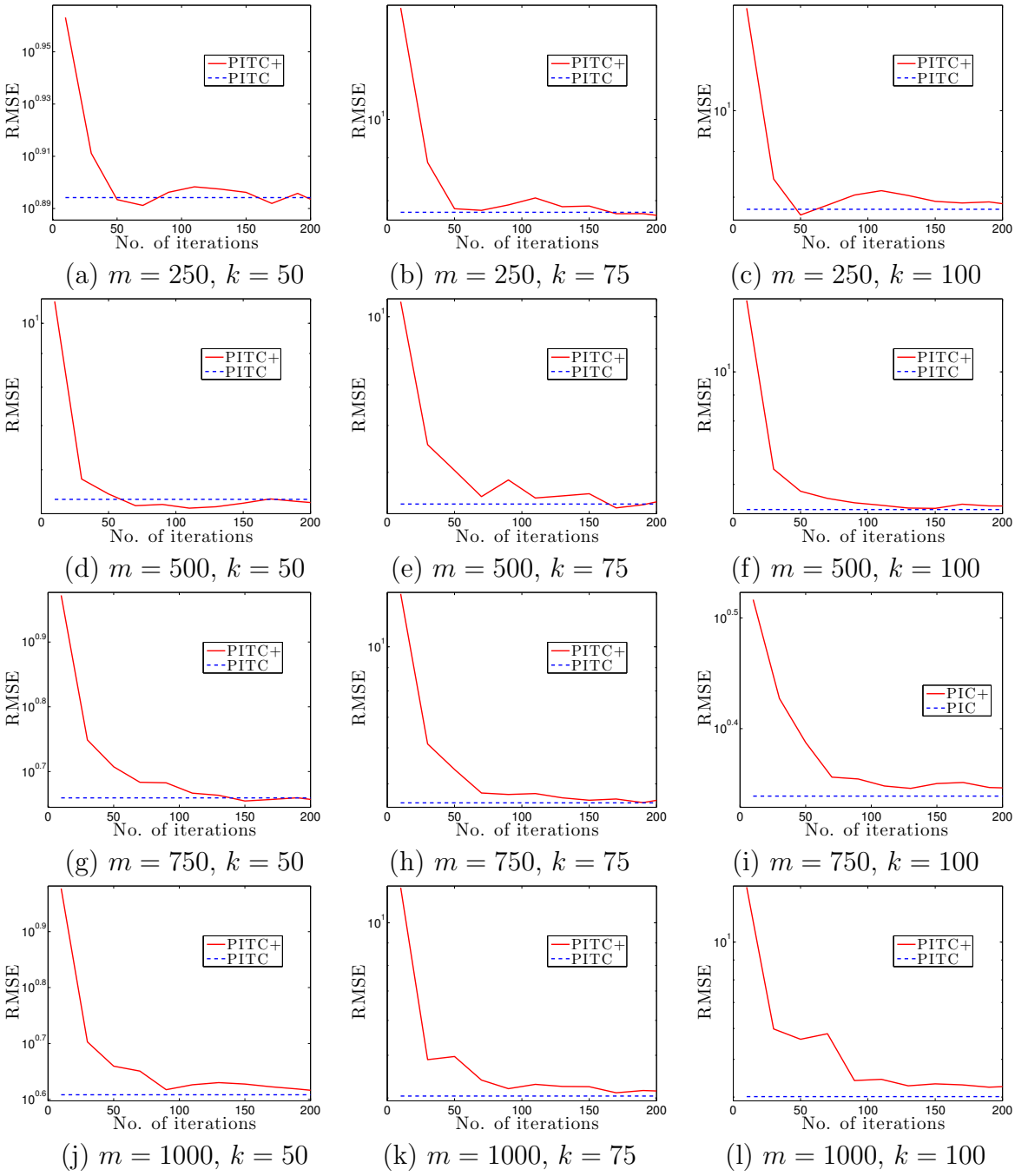
(a) $m = 250$, $k = 50$ (b) $m = 250$, $k = 75$ (c) $m = 250$, $k = 100$

(d) $m = 500$, $k = 50$ (e) $m = 500$, $k = 75$ (f) $m = 500$, $k = 100$

(g) $m = 750$, $k = 50$ (h) $m = 750$, $k = 75$ (i) $m = 750$, $k = 100$

(j) $m = 1000$, $k = 50$ (k) $m = 1000$, $k = 75$ (l) $m = 1000$, $k = 100$

Figure 5.6: PIC+'s anytime predictive performance on the SARCOS dataset with varying support set size $m = 250, 500, 750, 1000$ and number $k = 50, 75, 100$ of blocks.



(a) $m = 250$, $k = 50$ (b) $m = 250$, $k = 75$ (c) $m = 250$, $k = 100$

(d) $m = 500$, $k = 50$ (e) $m = 500$, $k = 75$ (f) $m = 500$, $k = 100$

(g) $m = 750$, $k = 50$ (h) $m = 750$, $k = 75$ (i) $m = 750$, $k = 100$

(j) $m = 1000$, $k = 50$ (k) $m = 1000$, $k = 75$ (l) $m = 1000$, $k = 100$

Figure 5.7: PITC+'s anytime predictive performance on the SARCOS dataset with varying support set size $m = 250, 500, 750, 1000$ and number $k = 50, 75, 100$ of blocks.

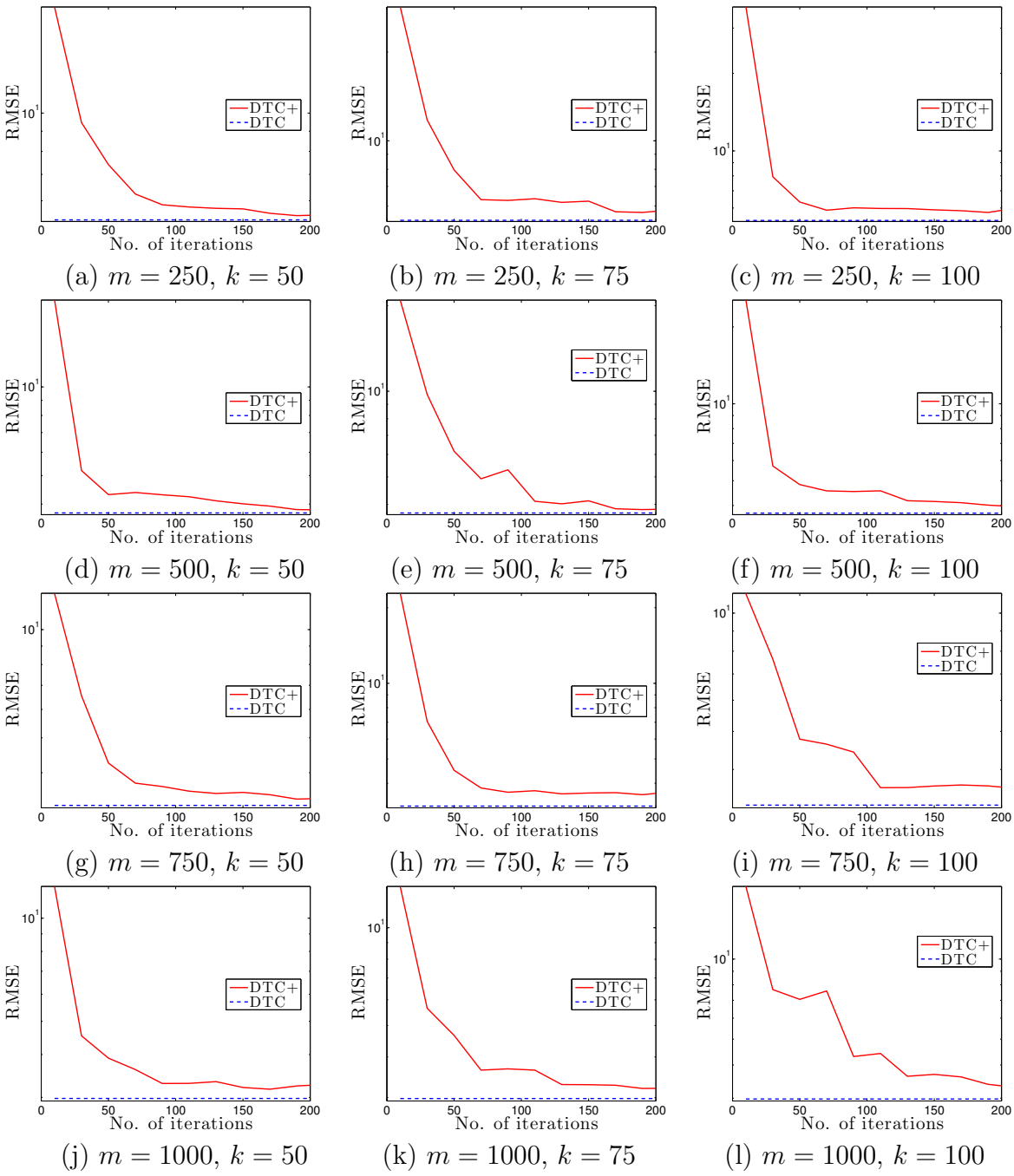
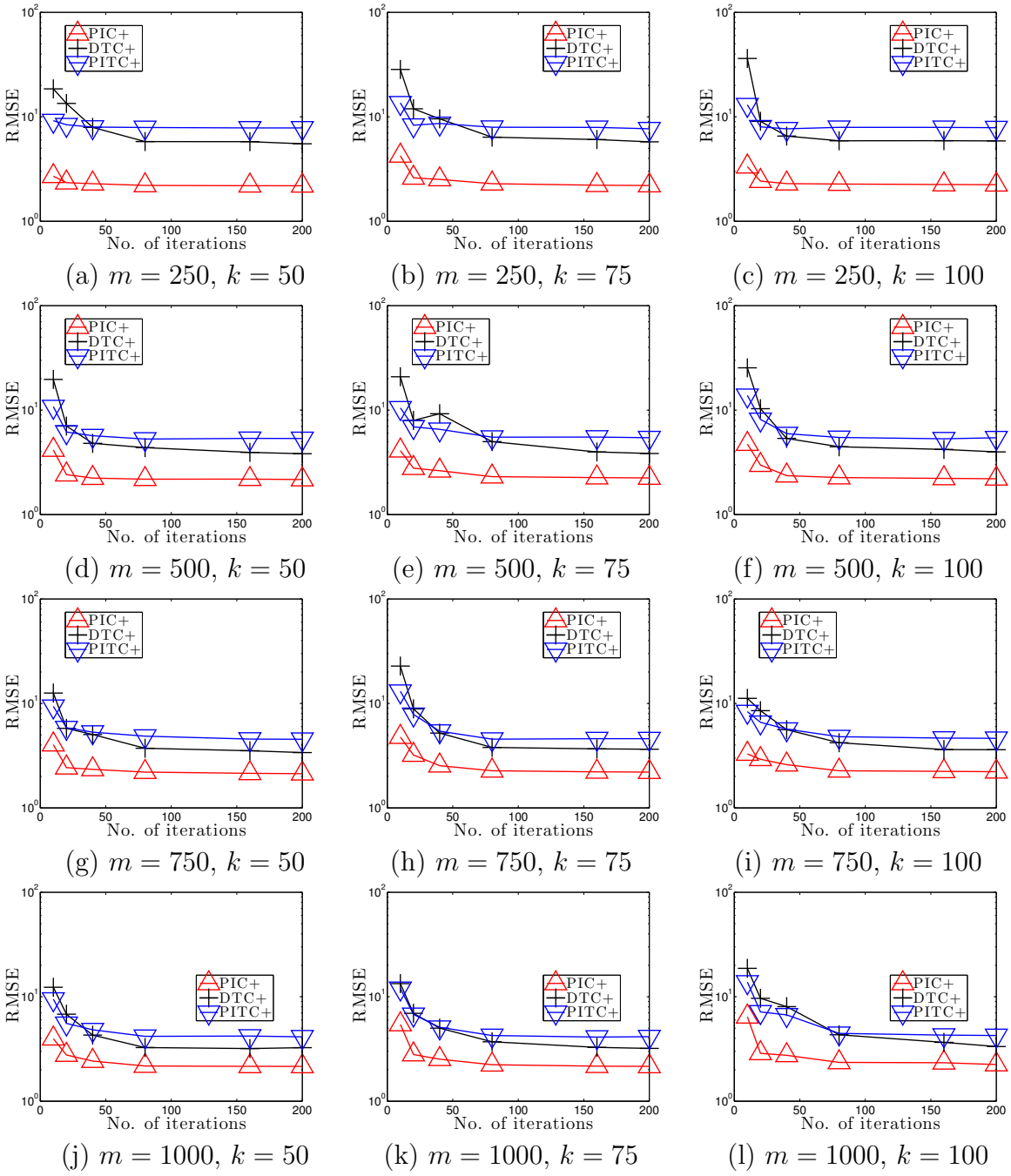


Figure 5.8: DTC+'s anytime predictive performance on the SARCOS dataset with varying support set size $m = 250, 500, 750, 1000$ and number $k = 50, 75, 100$ of blocks.



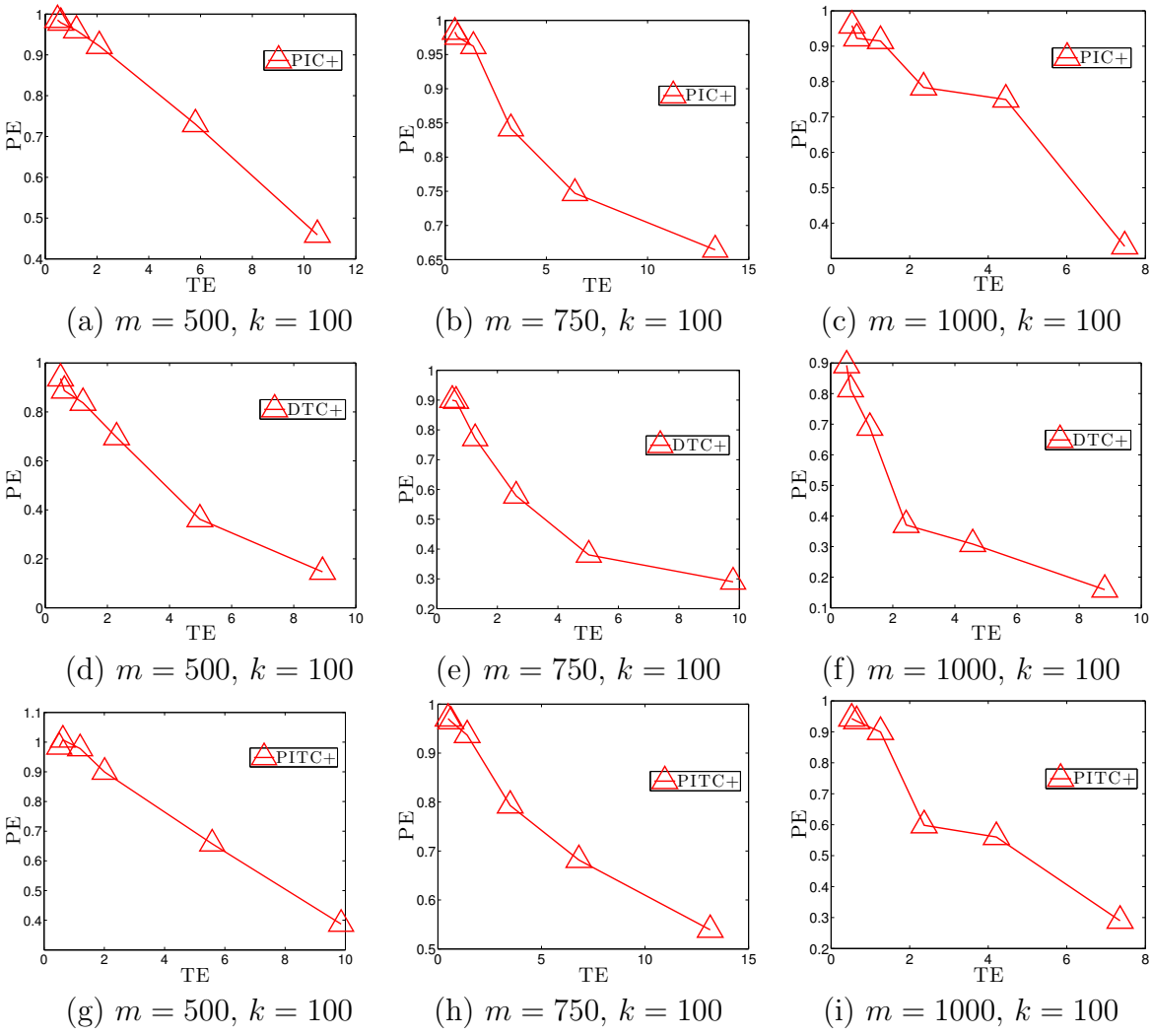
(a) $m = 250$, $k = 50$ (b) $m = 250$, $k = 75$ (c) $m = 250$, $k = 100$

(d) $m = 500$, $k = 50$ (e) $m = 500$, $k = 75$ (f) $m = 500$, $k = 100$

(g) $m = 750$, $k = 50$ (h) $m = 750$, $k = 75$ (i) $m = 750$, $k = 100$

(j) $m = 1000$, $k = 50$ (k) $m = 1000$, $k = 75$ (l) $m = 1000$, $k = 100$

Figure 5.9: Graphs of the anytime RMSE of PIC+, PITC+ and DTC+ evaluated on the SARCOS dataset with varying support set size $m = 250, 500, 750, 1000$ and number $k = 50, 75, 100$ of blocks.

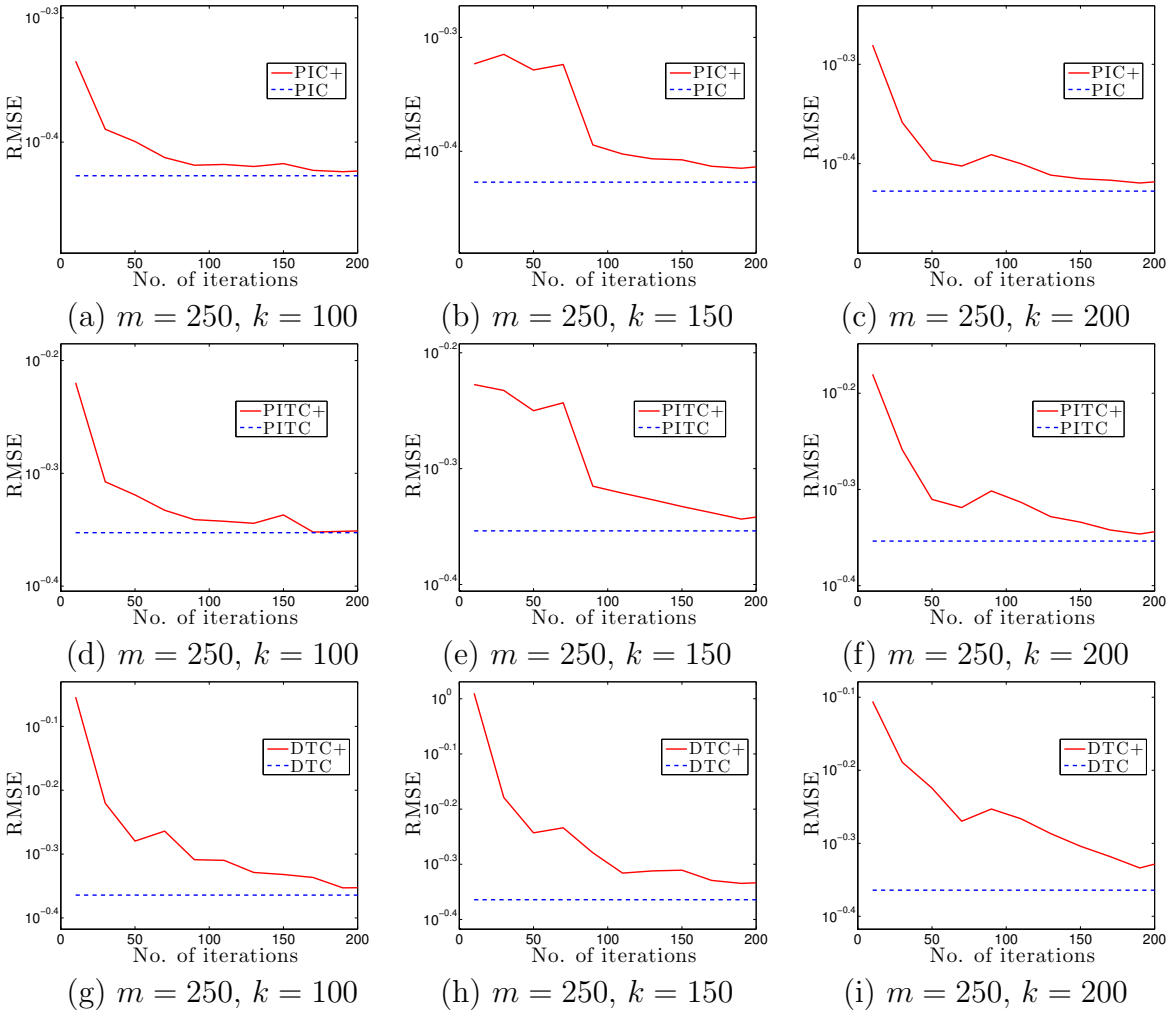


(a) $m = 500$, $k = 100$ (b) $m = 750$, $k = 100$ (c) $m = 1000$, $k = 100$

(d) $m = 500$, $k = 100$ (e) $m = 750$, $k = 100$ (f) $m = 1000$, $k = 100$

(g) $m = 500$, $k = 100$ (h) $m = 750$, $k = 100$ (i) $m = 1000$, $k = 100$

Figure 5.10: Graphs of time vs. prediction efficiency (TE vs. PE) trade-off for (a-c) PIC+, (d-f) DTC+ and (g-i) PITC+ evaluated on the SARCOS dataset with varying support set size $m = 500, 750, 1000$ and number $k = 100$ of blocks.

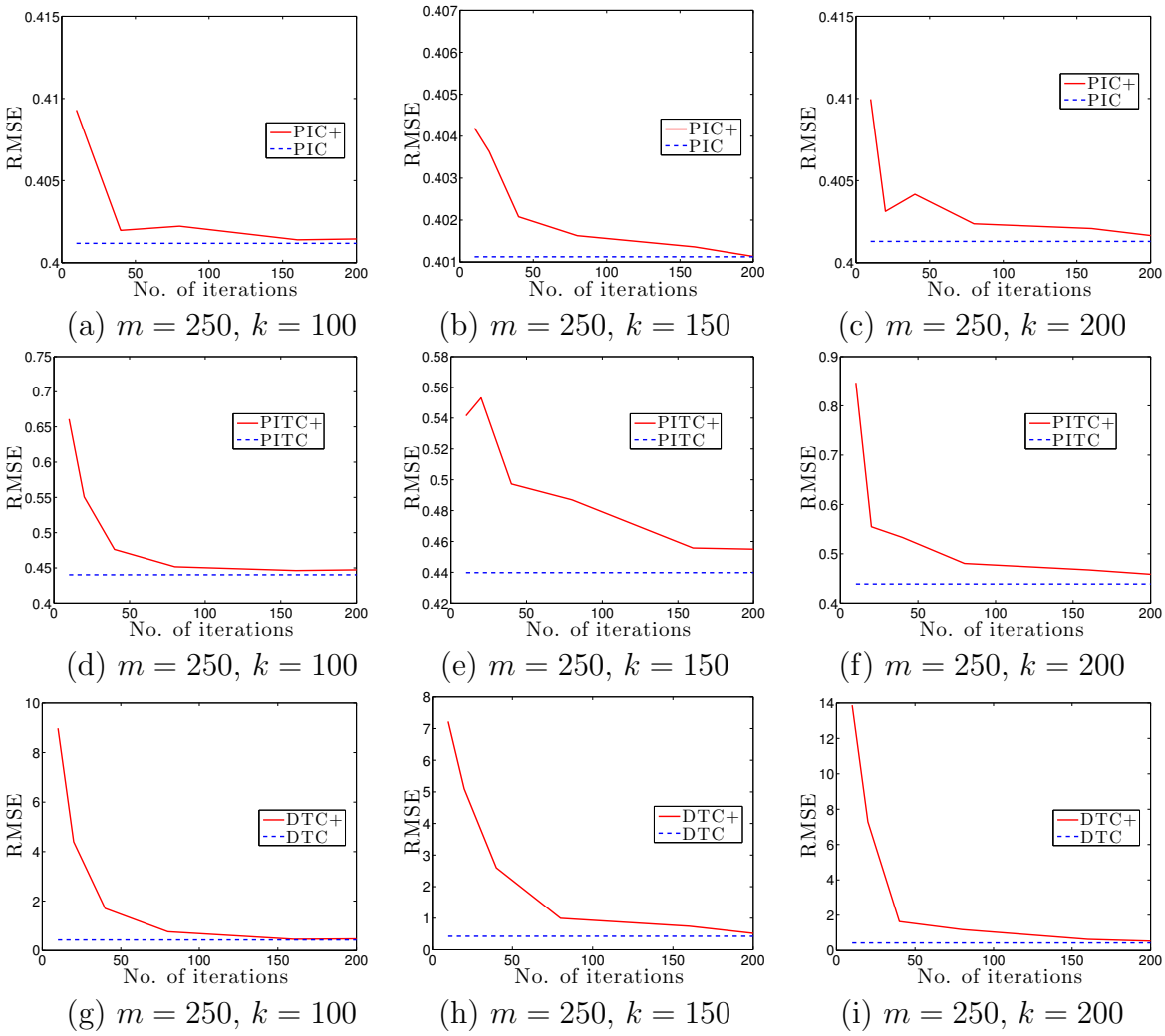


(a) $m = 250$, $k = 100$ (b) $m = 250$, $k = 150$ (c) $m = 250$, $k = 200$

(d) $m = 250$, $k = 100$ (e) $m = 250$, $k = 150$ (f) $m = 250$, $k = 200$

(g) $m = 250$, $k = 100$ (h) $m = 250$, $k = 150$ (i) $m = 250$, $k = 200$

Figure 5.11: PIC+, PITC+ and DTC+'s anytime predictive performance converge towards those of PIC, PITC and DTC on the UK Housing Price dataset for flat apartments with support set size $m = 250$ and varying number $k = 100, 150, 200$ of blocks.

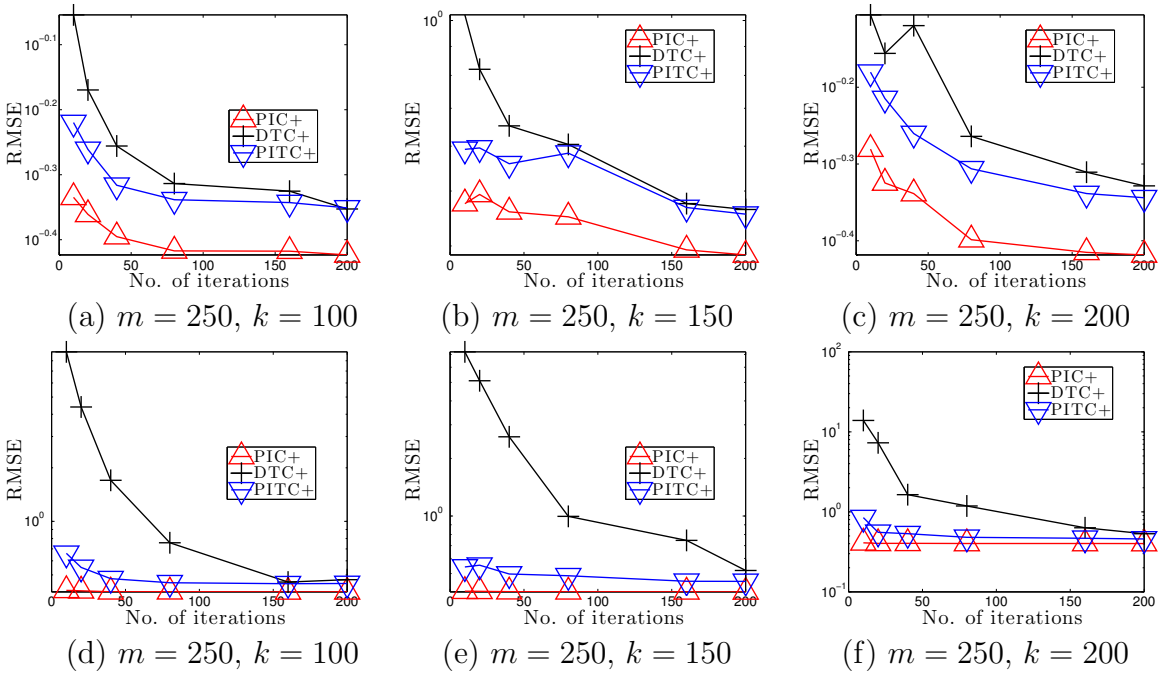


(a) $m = 250$, $k = 100$ (b) $m = 250$, $k = 150$ (c) $m = 250$, $k = 200$

(d) $m = 250$, $k = 100$ (e) $m = 250$, $k = 150$ (f) $m = 250$, $k = 200$

(g) $m = 250$, $k = 100$ (h) $m = 250$, $k = 150$ (i) $m = 250$, $k = 200$

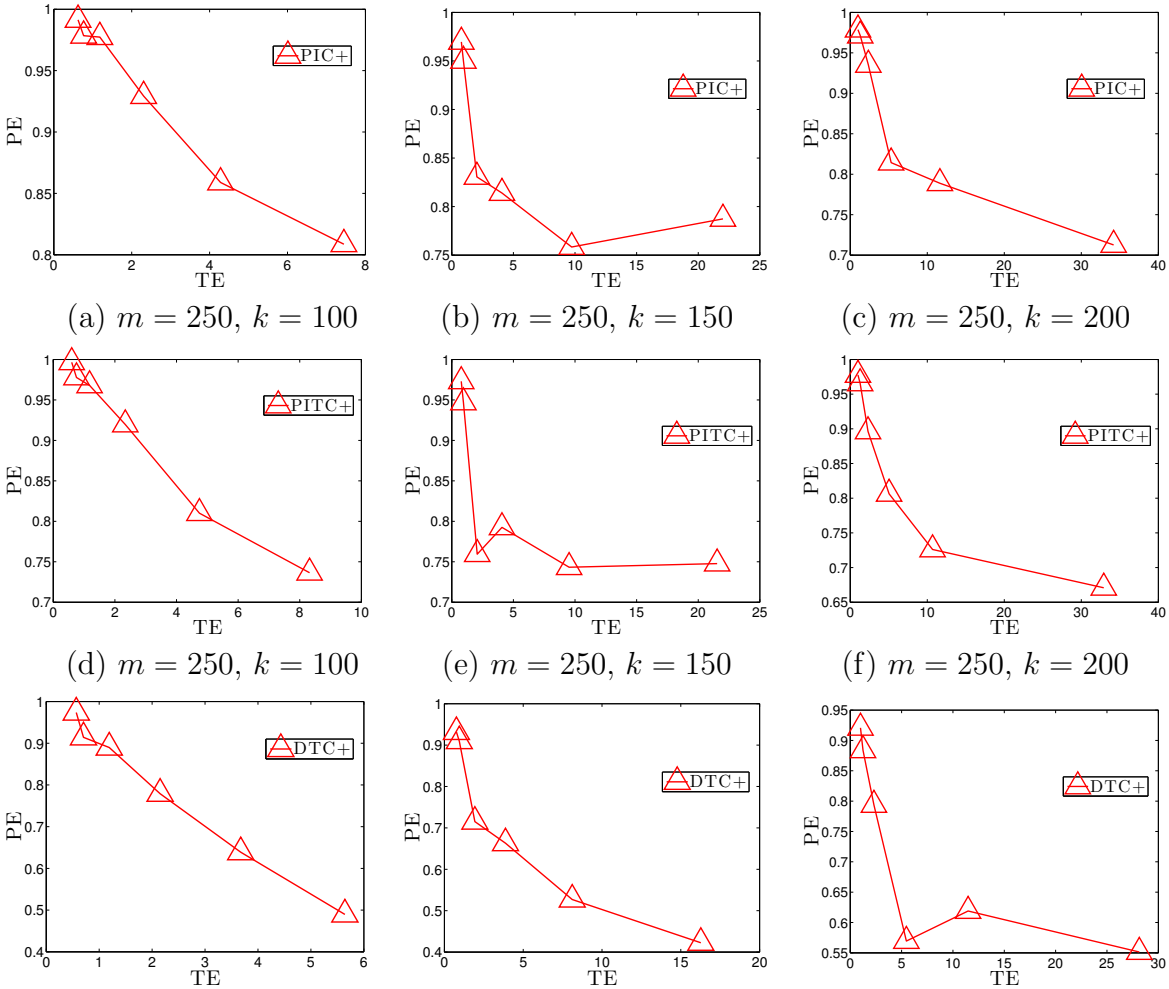
Figure 5.12: PIC+, PITC+ and DTC+'s anytime predictive performance converge towards those of PIC, PITC and DTC on the UK Housing Price dataset for detached houses with support set size $m = 250$ and varying number $k = 100, 150, 200$ of blocks.



(a) $m = 250$, $k = 100$ (b) $m = 250$, $k = 150$ (c) $m = 250$, $k = 200$

(d) $m = 250$, $k = 100$ (e) $m = 250$, $k = 150$ (f) $m = 250$, $k = 200$

Figure 5.13: Graphs of the anytime RMSE of PIC+, PITC+ and DTC+ evaluated on the UK Housing Price dataset for (a-c) flat apartments and (d-f) detached houses with support set size $m = 250$ and varying number $k = 100, 150, 200$ of blocks.

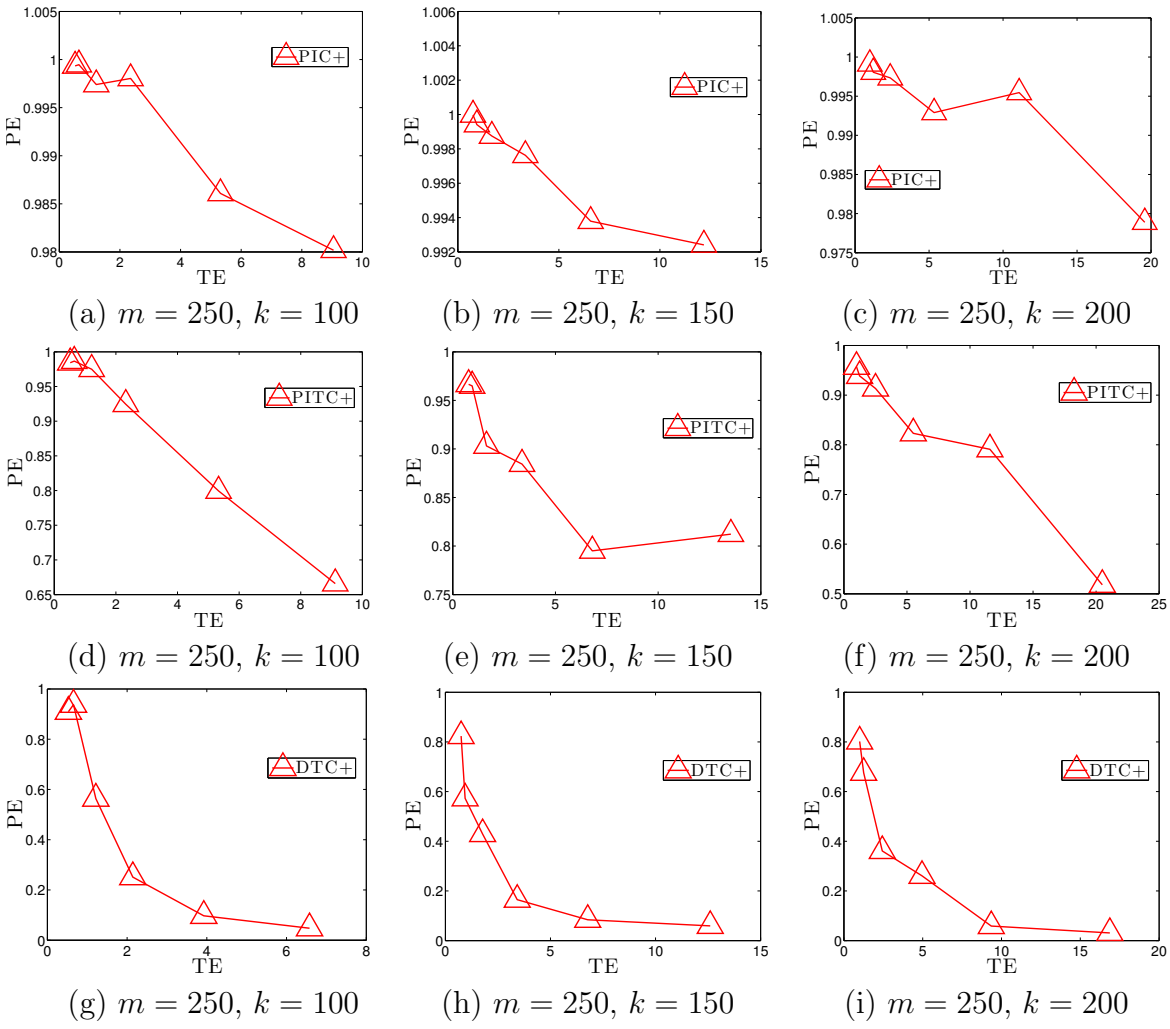


(a) $m = 250$, $k = 100$ (b) $m = 250$, $k = 150$ (c) $m = 250$, $k = 200$

(d) $m = 250$, $k = 100$ (e) $m = 250$, $k = 150$ (f) $m = 250$, $k = 200$

(g) $m = 250$, $k = 100$ (h) $m = 250$, $k = 150$ (i) $m = 250$, $k = 200$

Figure 5.14: Graphs of time vs. prediction efficiency (TE vs. PE) trade-off for (a-c) PIC+, (d-f) PITC+ and (g-i) DTC+ evaluated on the UK Housing Price dataset for flat apartments with support set size $m = 250$ and varying number $k = 100, 150, 200$ of blocks.



(a) $m = 250$, $k = 100$ (b) $m = 250$, $k = 150$ (c) $m = 250$, $k = 200$

(d) $m = 250$, $k = 100$ (e) $m = 250$, $k = 150$ (f) $m = 250$, $k = 200$

(g) $m = 250$, $k = 100$ (h) $m = 250$, $k = 150$ (i) $m = 250$, $k = 200$

Figure 5.15: Graphs of time vs. prediction efficiency (TE vs. PE) trade-off for (a-c) PIC+, (d-f) PITC+ and (g-i) DTC+ evaluated on the UK Housing Price dataset for detached houses with support set size $m = 250$ and varying number $k = 100, 150, 200$ of blocks.

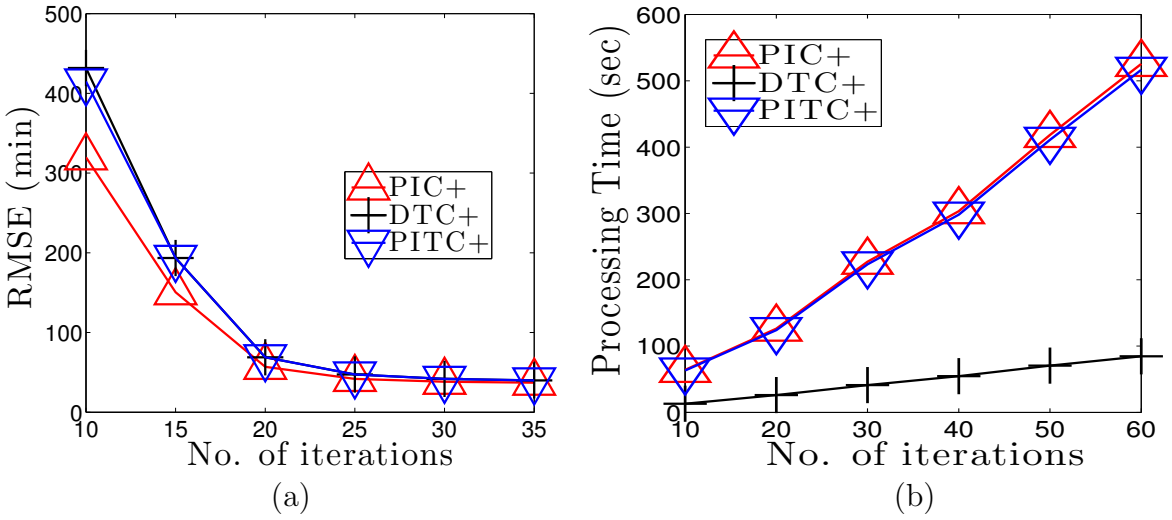


Figure 5.16: Graphs of (a) the anytime RMSE of PIC+, PITC+ and DTC+ evaluated on the AIRLINE dataset along with (b) their processing time with respect to $m = 1000$ supporting points and $k = 1000$ blocks.

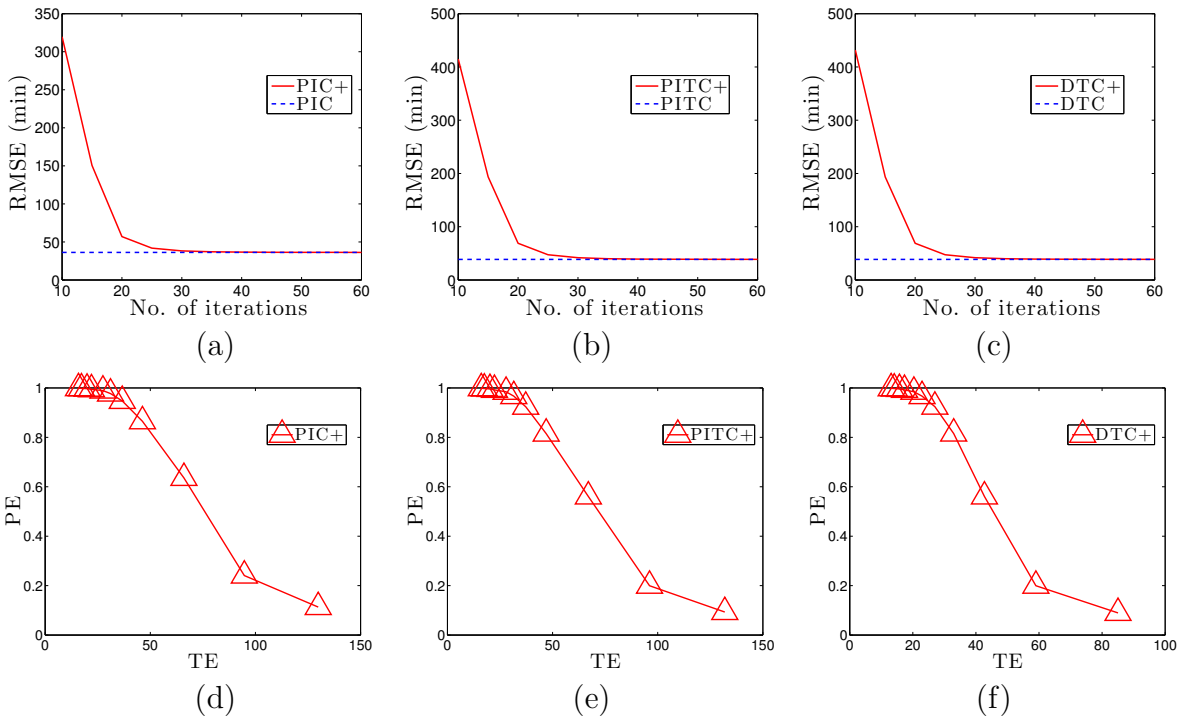


Figure 5.17: PIC+'s (a), PITC+'s (b) and DTC+'s (c) anytime prediction error empirically converges towards those of PIC, PITC and DTC on the AIRLINE dataset, and graphs of time vs. prediction efficiency trade-off for PIC+ (d), PITC+ (e) and DTC+ (f) with $m = 1000$ supporting points and $k = 1000$ blocks.

Chapter 6

Conclusion

This thesis has investigated the following question:

> Given a resource-constrained budget for interaction, how then does an interactive learning agent optimize the trade off between exploration and exploitation in practical, complex environmental domains efficiently?

6.1 Summary of Contributions

While working toward a satisfactory answer to the above question, along with practical algorithms that achieve it, we have been able to make the following progress:

> We have generalized the existing BRL framework [Poupart *et al.*, 2006] to integrate the general class of parametric models and model priors of the environment, thus successfully bridging the gap in applying BRL to more realistic and practical problem domains such as self-interested multi-agent learning [Hoang and Low, 2013a].
>
> We have established a formal, nonmyopic framework to circumvent this exploration-exploitation dilemma in budgeted AL scenarios, which guarantees a near

Bayes-optimal expected performance and consequently, closes up the gap in putting AL into practical, complex environmental domains while preserving the Bayes optimality [Hoang *et al.*, 2014].

To assist future developments of nonmyopic AL on large-scale environments, we have laid a theoretical foundation for scaling up the existing class of learning models to process massive datasets containing hundreds of thousands data points on a single machine. Though it is difficult to foresee the future of nonmyopic AL research in large-scale domains, its trajectory will likely require more scalable learning models.

All of which are substantiated by the following specific contributions:

- Formalization of I-BRL. I-BRL significantly extends BRL to integrate the general class of parametric models and model priors of the environment (Section 3.1) and consequently, relaxes the restrictive assumption of BRL that is often imposed in existing works and offers practitioners greater flexibility to encode their prior domain knowledge effectively.

- Solving I-BRL. Through I-BRL, it is demonstrated that the nonmyopic Bayes-optimal policy can be analytically derived (Section 3.2.1) and efficiently approximated (Section 3.2.2) with respect to an arbitrary choice of model and model prior for the unknown environment. Then, in practice, we empirically show the effectiveness of I-BRL in an interesting traffic problem modeled after a real-world situation for which the restrictive assumptions of BRL do not hold (Section 3.3).

- Formalization of ϵ-BAL. To establish a theoretical foundation for trading off between exploration and exploitation in nonmyopic active learning, which is still a research topic in its infancy, we develop a novel ϵ-BAL learning paradigm that

frames active learning as a Bayesian sequential decision problem to jointly and naturally optimize the exploration-exploitation trade-off while preserving the desired Bayes optimality (Section 4.2.1). Using ϵ-BAL, we can recognize and then, penalize a policy that biases towards exploitation if it entails a highly dispersed posterior over the model parameters. Consequently, the induced policy is guaranteed to be optimal in the expected active learning performance.

- Solving ϵ-BAL. Although the exact Bayes-optimal policy to nonmyopic active learning cannot be derived exactly, ϵ-BAL demonstrates that it is nevertheless possible to solve for an ϵ-Bayes-optimal policy analytically (Sections 4.2.2 and 4.2.3) and approximate it efficiently using an anytime algorithm based on ϵ-BAL with real-time performance guarantees (Section 4.2.4). In practice, we evaluate and verify its superior performance over the existing state-of-the-art algorithms using both synthetic and real-world datasets (Section 4.3).

- Scaling up Sparse Gaussian Processes (SGPs) for Big Data. To scale up the existing SGP-based learning models to large datasets, we demonstrate how a numerical approximation procedure which converges towards a given SGP model can be derived by a novel inverse variational inference framework (Section 5.2.1). Interestingly, we identify a class of SGP-based models for which it is possible to make the complexity for each update iteration independent of the size of data, thus resulting in an anytime learning paradigm that naturally trades off between the computing resource and the accuracy of estimation (Section 5.2.2). In practice, we demonstrate the efficiency of our framework on a wide range of large-scale real-world datasets which contain hundreds of thousand data points (Section 5.3).

6.2 Future Works

This section proposes and discusses potential research directions that could be pursued as continuation to our current work in this thesis:

- Large-scale Nonmyopic Active Learning. The work in this thesis has mainly focused on small-scale AL applications for which the learning agent is constrained by an active budget of at most a few hundreds of experiments/queries where each of which returns a single observation sample, thus underrating the scalability of their underlying predictive model (e.g., Gaussian processes). In contrast, for data-intensive domains (see Chapter 2.2), each conducted experiment/query might return a batch of observations instead of a single one, thus tremendously increasing the size of the collected dataset. This consequently renders the predictive model computationally impractical. For such reason, it is therefore highly interesting and desirable to develop AL algorithms capable of simultaneously selecting a batch of experiments/queries per stage while optimizing the exploration-exploitation trade-off in such nonmyopic fashion.

  Although we have already developed more scalable learning models to assist future developments in this direction (Chapter 5), deriving scalable, nonmyopic AL strategies to perform efficiently in large-scale domains remains highly nontrivial. In fact, a simple approach toward achieving this is to fix the batch size k in advance and exhaustively enumerate all possible combinations of k actions at each decision-making stage of the single-mode AL algorithm introduced in Chapter 4. However, this is highly inefficient in time-critical applications as the cost of enumerating these combinations will certainly grow exponentially in k. To avoid this computational bottleneck, one feasible approach is to consider using the developed nonmyopic AL policy (Chapter 4) on a *macro* level which

sequentially directs the learning agent towards the most informative regions (given its previously collected data) and uses the existing greedy (single-mode) AL strategies to quickly select a batch of observations. Analyzing the theoretical performance of this *macro* approach is an important step to gauge its feasibility in practice, which could be pursued as continuation to our work.

- Multi-output Nonmyopic Active Learning. The existing AL literature, including the work in this thesis, is still restricted to single-output learning scenarios for which each measurement is a single scalar. In practice, however, it is more often that we encounter multi-output domains where each measurement is a vector of multiple components which are probably correlated: Treating them separately essentially means neglecting important information which might lead to poor prediction. More specifically, these measurements may only be partially observable to us, meaning that parts of such measurement vectors are not directly observable to us. This further raises the incomplete/missing data issue. Assuming we have plenty of data for one particular component but significantly less for another, can we then exploit the correlation between these components to improve our prediction for the *less-data* component?

  Naively, this involves constructing an inter-correlation structure to model this phenomenon and facilitate a full Bayesian treatment. However, while there exists a multitude of works [Higdon, 2002; Boyle and Frean, 2005; Bonilla *et al.*, 2008; Álvarez *et al.*, 2010] addressing this problem, it is still unclear whether the proposed multi-output models indeed exhibit a necessary decomposable structure (see Eqs. (5.20) and (5.21)) which could lend itself to the development of a similar anytime strategy as we previously demonstrated in Chapter 5 for

the single-output case. Ultimately, how do we integrate this in a nonmyopic AL framework efficiently? These are the highly non-trivial questions that we seek to address as a potential extension of our current work.

Appendix A

Proofs of Main Results for Chapter 3

This section provides more detailed proof sketches for Theorems 2 and 3 as mentioned in Section 3.2.

A.1 Proof of Theorem 2

Theorem 2. The optimal value function V^k for k steps-to-go converges to the optimal value function V for infinite horizon as $k \to \infty$:

$$\|V - V^{k+1}\|_\infty \;\leq\; \phi\|V - V^k\|_\infty \,. \tag{A.1}$$

Proof Sketch. Define $L_s^k(b) = |V_s(b) - V_s^k(b)|$. Using $|\max_a f(a) - \max_a g(a)| \leq \max_a |f(a) - g(a)|$,

$$\begin{aligned} L_s^{k+1}(b) &\leq \phi \max_u \sum_{v,s'} \langle p_s^v, b\rangle \, p_s^{uv}(s') L_{s'}^k(b_s^v) \\ &\leq \phi \max_u \sum_{v,s'} \langle p_s^v, b\rangle \, p_s^{uv}(s') \|V - V^k\|_\infty \\ &= \phi\|V - V^k\|_\infty \,. \end{aligned} \tag{A.2}$$

Since the last inequality (A.2) holds for every pair (s, b), it follows that $\|V - V^{k+1}\|_\infty \leq \phi\|V - V^k\|_\infty$. This completes our proof. $\square$

A.2 Proof of Theorem 3

Theorem 3. The optimal value function $V_s^k(b)$ for k steps-to-go can be represented as a finite set Γ_s^k of α-functions:

$$V_s^k(b) = \max_{\alpha_s \in \Gamma_s^k} \langle \alpha_s, b \rangle \ . \tag{A.3}$$

Proof Sketch. We give a constructive proof to (A.3) by induction, which shows how Γ_s^k can be built recursively. Assuming that (A.3) holds for k [1], it can be proven that (A.3) also holds for $k+1$. In particular, it follows from our inductive assumption that the term $V_{s'}^k(b_s^v)$ in (3.5) can be rewritten as:

$$\begin{aligned} V_{s'}^k(b_s^v) &= \max_{j=1}^{|\Gamma_{s'}^k|} \int_\lambda \alpha_{s'}^j(\lambda) b_s^v(\lambda) \mathrm{d}\lambda \\ &= \max_{j=1}^{|\Gamma_{s'}^k|} \int_\lambda \alpha_{s'}^j(\lambda) \frac{p_s^v(\lambda) b(\lambda)}{\langle p_s^v, b \rangle} \mathrm{d}\lambda \\ &= \langle p_s^v, b \rangle^{-1} \max_{j=1}^{|\Gamma_{s'}^k|} \int_\lambda b(\lambda) \alpha_{s'}^j(\lambda) p_s^v(\lambda) \mathrm{d}\lambda \ . \end{aligned} \tag{A.4}$$

By plugging the above equation into (3.5) and using $r_b^s(u) = \sum_v \langle p_s^v, b \rangle \, r_s(u, v)$,

$$V_s^{k+1}(b) = \max_u \left(r_b^s(u) + \phi \sum_{s', v} \max_{j=1}^{|\Gamma_{s'}^k|} p_s^{uv}(s') Q_s^v(\alpha_{s'}^j, b) \right) \ , \tag{A.5}$$

[1] When $k = 0$, (A.3) can be verified by letting $\alpha_s(\lambda) = 0$

where $Q_s^v(\alpha_{s'}^j, b) = \int_\lambda \alpha_{s'}^j(\lambda) p_s^v(\lambda) b(\lambda) \mathrm{d}\lambda$. Now, applying the fact that

$$\sum_{s'} \sum_{v} \max_{t_{s'v}=1}^{|\Gamma_{s'}^k|} A_{s'v}[t_{s'v}] \;=\; \max_{t} \sum_{s'} \sum_{v} A_{s'v}[t_{s'v}] \;, \tag{A.6}$$

where $A_{s'v}[t_{s'v}] = p_s^{uv}(s') Q_s^v(\alpha_{s'}^{t_{s'v}}, b)$ and using $r_b^s(u) = \int_\lambda b(\lambda) \sum_v p_s^v(\lambda) r_s(u,v) \mathrm{d}\lambda$, equation (A.5) can be rewritten as

$$V_s^{k+1}(b) \;=\; \max_{u,t} \int_\lambda b(\lambda) \alpha_s^{ut}(\lambda) \mathrm{d}\lambda \;, \tag{A.7}$$

with $t = (t_{s'v})_{s' \in S, v \in V}$ and

$$\alpha_s^{ut}(\lambda) \;=\; \sum_v p_s^v(\lambda) \left(r_s(u,v) + \phi \sum_{s'} \alpha_{s'}^{t_{s'v}}(\lambda) p_s^{uv}(s') \right) \;. \tag{A.8}$$

By setting $\Gamma_s^{k+1} = \{\alpha_s^{ut}\}_{u,t}$ and $u_{\alpha_s^{ut}} = u$, it can be verified that (A.3) also holds for $k+1$. Our proof is therefore completed. $\square$

Appendix B

Proofs of Main Results for Chapter 4

B.1 Proof of Lemma 1

We will give a proof by induction on n that

$$|Q_n^*(z_{\mathcal{D}}, x) - Q_n^\epsilon(z_{\mathcal{D}}, x)| \;\leq\; (N - n + 1)\gamma \tag{B.1}$$

for all tuples $(n, z_{\mathcal{D}}, x)$ generated at stage $n = n', \ldots, N$ by (4.9) to compute $V_{n'}^\epsilon(z_{\mathcal{D}'})$. When $n = N$, $W_N^*(z_{\mathcal{D}}, x) = Q_N^\epsilon(z_{\mathcal{D}}, x)$ in (4.11), by definition. So, $|Q_N^*(z_{\mathcal{D}}, x) - Q_N^\epsilon(z_{\mathcal{D}}, x)| \leq \gamma$ (B.1) trivially holds for the base case. Supposing (B.1) holds for $n+1$ (i.e., induction hypothesis), we will prove that it holds for $n' \leq n < N$:

$$\begin{aligned} |Q_n^*(z_{\mathcal{D}}, x) - Q_n^\epsilon(z_{\mathcal{D}}, x)| &\leq |Q_n^*(z_{\mathcal{D}}, x) - W_n^*(z_{\mathcal{D}}, x)| + |W_n^*(z_{\mathcal{D}}, x) - Q_n^\epsilon(z_{\mathcal{D}}, x)| \\ &\leq \gamma + |W_n^*(z_{\mathcal{D}}, x) - Q_n^\epsilon(z_{\mathcal{D}}, x)| \\ &\leq \gamma + (N - n)\gamma = (N - n + 1)\gamma \;. \end{aligned} \tag{B.2}$$

The first and second inequalities follow from the triangle inequality and (4.11), respectively. The last inequality is due to

$$\begin{aligned} |W_n^*(z_{\mathcal{D}}, x) - Q_n^\epsilon(z_{\mathcal{D}}, x)| &\leq \frac{1}{S}\sum_{i=1}^{S} \left|V_{n+1}^*(z_{\mathcal{D}} \cup \{z_x^i\}) - V_{n+1}^\epsilon(z_{\mathcal{D}} \cup \{z_x^i\})\right| \\ &\leq \frac{1}{S}\sum_{i=1}^{S} \max_{x'} \left|Q_{n+1}^*(z_{\mathcal{D}} \cup \{z_x^i\}, x') - Q_{n+1}^\epsilon(z_{\mathcal{D}} \cup \{z_x^i\}, x')\right| \\ &\leq (N-n)\gamma \end{aligned} \tag{B.3}$$

such that the last inequality follows from the induction hypothesis. From (B.1), when $n = n'$, $|Q_{n'}^*(z_{\mathcal{D}'}, x) - Q_{n'}^\epsilon(z_{\mathcal{D}'}, x)| \leq (N - n' + 1)\gamma$ for all $x \in \mathcal{X} \setminus \mathcal{D}'$ since $\mathcal{D} = \mathcal{D}'$ and $z_{\mathcal{D}} = z_{\mathcal{D}'}$. □

B.2 Proof of Lemma 2

Let's define $W_n^i(z_{\mathcal{D}}, x) \triangleq -\log p(z_x^i|z_{\mathcal{D}}) + V_{n+1}^*(z_{\mathcal{D}} \cup \{z_x^i\})$. Then, $W_n^*(z_{\mathcal{D}}, x) = S^{-1}\sum_{i=1}^{S} W_n^i(z_{\mathcal{D}}, x)$ can be viewed as an empirical mean computed based on the random samples $W_n^i(z_{\mathcal{D}}, x)$ drawn from a distribution whose mean coincides with

$$\begin{aligned} \widehat{Q}_n\left(z_{\mathcal{D}}, x\right) &\triangleq \widehat{\mathbb{H}}\left[\widehat{Z}_x|z_{\mathcal{D}}\right] + \mathbb{E}\left[V_{n+1}^*\left(z_{\mathcal{D}} \cup \{\widehat{Z}_x\}\right)|z_{\mathcal{D}}\right] \\ \widehat{\mathbb{H}}\left[\widehat{Z}_x|z_{\mathcal{D}}\right] &\triangleq -\int_{-\widehat{\tau}}^{\widehat{\tau}} f\left(\widehat{Z}_x = z_x|z_{\mathcal{D}}\right) \log p\left(Z_x = z_x|z_{\mathcal{D}}\right) \mathrm{d}z_x \\ &- f\left(\widehat{Z}_x = -\widehat{\tau}|z_{\mathcal{D}}\right) \log p\left(Z_x = -\widehat{\tau}|z_{\mathcal{D}}\right) \\ &- f\left(\widehat{Z}_x = \widehat{\tau}|z_{\mathcal{D}}\right) \log p\left(Z_x = \widehat{\tau}|z_{\mathcal{D}}\right) \end{aligned} \tag{B.4}$$

such that the expectation term is omitted from the RHS expression of $\widehat{Q}_N$ at stage N, and recall from Definition 1 that f and p are distributions of $\widehat{Z}_x$ and Z_x, respectively.

Using Hoeffding's inequality,

$$\left| \widehat{Q}_n\left(z_D, x\right) - \frac{1}{S}\sum_{i=1}^{S} W_n^i\left(z_D, x\right) \right| \;\leq\; \frac{\gamma}{2}$$

with probability at least $1 - 2\exp(-S\gamma^2/\left(2(\overline{W} - \underline{W})^2\right))$ where $\overline{W}$ and $\underline{W}$ are upper and lower bounds of $W_n^i(z_{\mathcal{D}}, x)$, respectively. To determine these bounds, note that $|z_x^i| \leq \widehat{\tau}$, by Definition 1, and $|\mu_{x|\mathcal{D},\lambda}| \leq \widehat{\tau} - \tau$, by (4.8). Consequently, $0 \leq (z_x^i - \mu_{x|\mathcal{D},\lambda})^2 \leq (2\widehat{\tau} - \tau)^2 \leq (2N\kappa^{N-1}\tau - \tau)^2 = (2N\kappa^{N-1} - 1)^2\tau^2$ such that the last inequality follows from Lemma 8. Together with using Lemma 7, the following result ensues:

$$\frac{1}{\sqrt{2\pi\sigma_n^2}} \;\geq\; p(z_x^i|z_{\mathcal{D}}, \lambda) \geq \frac{1}{\sqrt{2\pi\sigma_o^2}} \exp\left(\frac{-(2N\kappa^{N-1} - 1)^2\tau^2}{2\sigma_n^2}\right) \tag{B.5}$$

where σ_n^2 and σ_o^2 are previously defined in (4.14). It follows that

$$\begin{aligned} p(z_x^i|z_{\mathcal{D}}) &= \sum_{\lambda\in\Lambda} p(z_x^i|z_{\mathcal{D}}, \lambda)\; b_{\mathcal{D}}(\lambda) \\ &\geq \sum_{\lambda\in\Lambda} \left[\frac{1}{\sqrt{2\pi\sigma_o^2}} \exp\left(\frac{-(2N\kappa^{N-1} - 1)^2\tau^2}{2\sigma_n^2}\right)\right] b_{\mathcal{D}}(\lambda) \\ &= \frac{1}{\sqrt{2\pi\sigma_o^2}} \exp\left(\frac{-(2N\kappa^{N-1} - 1)^2\tau^2}{2\sigma_n^2}\right) . \end{aligned}$$

Similarly, $p(z_x^i|z_{\mathcal{D}}) \leq 1/\sqrt{2\pi\sigma_n^2}$. Then,

$$\begin{aligned} -\log p(z_x^i|z_{\mathcal{D}}) &\leq \frac{1}{2}\log\left(2\pi\sigma_o^2\right) + \frac{(2N\kappa^{N-1} - 1)^2\tau^2}{2\sigma_n^2} , \\ -\log p(z_x^i|z_{\mathcal{D}}) &\geq \frac{1}{2}\log\left(2\pi\sigma_n^2\right) . \end{aligned}$$

By Lemma 11, $0.5(N-n)\log\left(2\pi e\sigma_n^2\right) \leq V_{n+1}^*(z_{\mathcal{D}} \cup \{z_x^i\}) \leq 0.5(N-n)\log\left(2\pi e\sigma_o^2\right) + \log|\Lambda|$. Consequently,

$$\begin{aligned}\left|\overline{W} - \underline{W}\right| &\leq N\log\left(\frac{\sigma_o}{\sigma_n}\right) + \frac{(2N\kappa^{N-1}-1)^2\tau^2}{2\sigma_n^2} + \log|\Lambda| \\ &= \mathcal{O}\left(\frac{N^2\kappa^{2N}\tau^2}{\sigma_n^2} + N\log\frac{\sigma_o}{\sigma_n} + \log|\Lambda|\right).\end{aligned}$$

Finally, using Lemma 17, $|Q_n^*(z_{\mathcal{D}},x) - \widehat{Q}_n(z_{\mathcal{D}},x)| \leq \gamma/2$ by setting

$$\tau = \mathcal{O}\left(\sigma_o\sqrt{\log\left(\frac{\sigma_o^2}{\gamma}\left(\frac{N^2\kappa^{2N}+\sigma_o^2}{\sigma_n^2} + N\log\frac{\sigma_o}{\sigma_n} + \log|\Lambda|\right)\right)}\right),$$

thereby guaranteeing that

$$\begin{aligned}|Q_n^*(z_{\mathcal{D}},x) - W_n^*(z_{\mathcal{D}},x)| &\leq |Q_n^*(z_{\mathcal{D}},x) - \widehat{Q}_n(z_{\mathcal{D}},x)| + |\widehat{Q}_n(z_{\mathcal{D}},x) - W_n^*(z_{\mathcal{D}},x)| \\ &\leq \frac{\gamma}{2} + \frac{\gamma}{2} = \gamma\end{aligned}$$

with probability at least $1 - 2\exp\left(-2S\gamma^2/T^2\right)$ where

$$T = 2\left|\overline{W} - \underline{W}\right| = \mathcal{O}\left(\frac{N^2\kappa^{2N}\tau^2}{\sigma_n^2} + N\log\frac{\sigma_o}{\sigma_n} + \log|\Lambda|\right). \quad \square$$

B.3 Proof of Lemma 3

From Lemma 2,

$$P(|Q_n^*(z_{\mathcal{D}},x) - W_n^*(z_{\mathcal{D}},x)| > \gamma) \leq 2\exp\left(-\frac{2S\gamma^2}{T^2}\right)$$

for each tuple $(n, z_{\mathcal{D}}, x)$ generated at stage $n = n', \ldots, N$ by (4.9) to compute $V_{n'}^{\epsilon}(z_{\mathcal{D}'})$. Since there will be no more than $(S|\mathcal{X}|)^N$ tuples $(n, z_{\mathcal{D}}, x)$ generated at stage $n =$

$n', \ldots, N$ by (4.9) to compute $V^{\epsilon}_{n'}(z_{\mathcal{D}'})$, the probability that $|Q^*_n(z_{\mathcal{D}}, x) - W^*_n(z_{\mathcal{D}}, x)| > \gamma$ for some generated tuple $(n, z_{\mathcal{D}}, x)$ is at most $2(S|\mathcal{X}|)^N \exp(-2S\gamma^2/T^2)$ by applying the union bound. Lemma 3 then directly follows. □

B.4 Proof of Theorem 5

Suppose that a set $z_{\mathcal{D}}$ of observations, a budget of $N - n + 1$ sampling locations, $S \in \mathbb{Z}^+$, and $\gamma > 0$ are given. It follows immediately from Lemmas 1 and 3 that the probability of $|Q^*_n(z_{\mathcal{D}}, x) - Q^{\epsilon}_n(z_{\mathcal{D}}, x)| \leq N\gamma$ (4.12) holding for all $x \in \mathcal{X} \setminus \mathcal{D}$ is at least

$$1 \quad - \quad 2\left(S\,|\mathcal{X}|\right)^N \exp\left(-\frac{2S\gamma^2}{T^2}\right)$$

where T is previously defined in Lemma 2.

To guarantee that $|Q^*_n(z_{\mathcal{D}}, x) - Q^{\epsilon}_n(z_{\mathcal{D}}, x)| \leq N\gamma$ (4.12) holds for all $x \in \mathcal{X} \setminus \mathcal{D}_0$ with probability at least $1 - \delta$, the value of S to be determined must therefore satisfy the following inequality:

$$1 - 2\left(S\,|\mathcal{X}|\right)^N \exp\left(-\frac{2S\gamma^2}{T^2}\right) \quad \geq \quad 1 - \delta\ ,$$

which is equivalent to

$$S \quad \geq \quad \frac{T^2}{2\gamma^2}\left(N \log S + N \log |\mathcal{X}| + \log \frac{2}{\delta}\right). \tag{B.6}$$

Using the identity $\log S \leq \alpha S - \log \alpha - 1$ with an appropriate choice of $\alpha = \gamma^2/(NT^2)$,

the RHS expression of (B.6) can be bounded from above by

$$\frac{S}{2}+\frac{T^2}{2\gamma^2}\left(N\log\frac{N|\mathcal{X}|T^2}{e\gamma^2}+\log\frac{2}{\delta}\right).$$

Therefore, to satisfy (B.6), it suffices to determine the value of S such that the following inequality holds:

$$S \;\geq\; \frac{S}{2}+\frac{T^2}{2\gamma^2}\left(N\log\frac{N|\mathcal{X}|T^2}{e\gamma^2}+\log\frac{2}{\delta}\right)$$

by setting

$$S \;=\; \frac{T^2}{\gamma^2}\left(N\log\frac{N|\mathcal{X}|T^2}{e\gamma^2}+\log\frac{2}{\delta}\right) \tag{B.7}$$

where $T \triangleq \mathcal{O}\left(\frac{N^2\kappa^{2N}\tau^2}{\sigma_n^2}+N\log\frac{\sigma_o}{\sigma_n}+\log|\Lambda|\right)$ by further setting

$$\tau \;=\; \mathcal{O}\left(\sigma_o\sqrt{\log\left(\frac{\sigma_o^2}{\gamma}\left(\frac{N^2\kappa^{2N}+\sigma_o^2}{\sigma_n^2}+N\log\frac{\sigma_o}{\sigma_n}+\log|\Lambda|\right)\right)}\right),$$

as defined in Lemma 2 previously. By assuming σ_o, σ_n, $|\Lambda|$, N, κ, and $|\mathcal{X}|$ as constants, $\tau=\mathcal{O}(\sqrt{\log(1/\gamma)})$, thus resulting in $T=\mathcal{O}(\log(1/\gamma))$. Consequently, (B.7) can be reduced to

$$S \;=\; \mathcal{O}\left(\frac{\left(\log\left(\frac{1}{\gamma}\right)\right)^2}{\gamma^2}\log\left(\frac{\log\left(\frac{1}{\gamma}\right)}{\gamma\delta}\right)\right). \quad \square$$

B.5 Proof of Lemma 4

Theorem 5 implies that (a) $Q_n^*(z_{\mathcal{D}}, \pi_n^*(z_{\mathcal{D}})) - Q_n^*(z_{\mathcal{D}}, \pi_n^\epsilon(z_{\mathcal{D}})) \leq Q_n^*(z_{\mathcal{D}}, \pi_n^*(z_{\mathcal{D}})) - Q_n^\epsilon(z_{\mathcal{D}}, \pi_n^\epsilon(z_{\mathcal{D}})) + N\gamma$ and (b) $Q_n^*(z_{\mathcal{D}}, \pi_n^*(z_{\mathcal{D}})) - Q_n^\epsilon(z_{\mathcal{D}}, \pi_n^\epsilon(z_{\mathcal{D}})) \leq \max_{x \in \mathcal{X} \setminus \mathcal{D}} |Q_n^*(z_{\mathcal{D}}, x) - Q_n^\epsilon(z_{\mathcal{D}}, x)| \leq N\gamma$. By combining (a) and (b), $Q_n^*(z_{\mathcal{D}}, \pi_n^*(z_{\mathcal{D}})) - Q_n^*(z_{\mathcal{D}}, \pi_n^\epsilon(z_{\mathcal{D}})) \leq N\gamma + N\gamma = 2N\gamma$ holds with probability at least $1 - \delta$ by setting S and τ according to that in Theorem 5. □

B.6 Proof of Theorem 6

By Lemma 4, $Q_n^*(z_{\mathcal{D}}, \pi_n^*(z_{\mathcal{D}})) - Q_n^*(z_{\mathcal{D}}, \pi_n^\epsilon(z_{\mathcal{D}})) \leq 2N\gamma$ holds with probability at least $1 - \delta$. Otherwise, $Q_n^*(z_{\mathcal{D}}, \pi_n^*(z_{\mathcal{D}})) - Q_n^*(z_{\mathcal{D}}, \pi_n^\epsilon(z_{\mathcal{D}})) > 2N\gamma$ with probability at most δ. In the latter case,

$$\begin{aligned} Q_n^*(z_{\mathcal{D}}, \pi_n^*(z_{\mathcal{D}})) - Q_n^*(z_{\mathcal{D}}, \pi_n^\epsilon(z_{\mathcal{D}})) &\leq (N - n + 1) \log\left(\frac{\sigma_o}{\sigma_n}\right) + \log|\Lambda| \\ &\leq N \log\left(\frac{\sigma_o}{\sigma_n}\right) + \log|\Lambda| \end{aligned}$$

where the first inequality in (B.8) follows from (a) $Q_n^*(z_{\mathcal{D}}, \pi_n^*(z_{\mathcal{D}})) = V_n^*(z_{\mathcal{D}}) \leq 0.5(N - n + 1)\log(2\pi e\sigma_o^2) + \log|\Lambda|$, by Lemma 11, and (b)

$$\begin{aligned} Q_n^*\left(z_{\mathcal{D}}, \pi_n^\epsilon(z_{\mathcal{D}})\right) &= \mathbb{H}\left[Z_{\pi_n^\epsilon(z_{\mathcal{D}})} | z_{\mathcal{D}}\right] + \mathbb{E}\left[V_{n+1}^*\left(z_{\mathcal{D}} \cup \{Z_{\pi_n^\epsilon(z_{\mathcal{D}})}\}\right) | z_{\mathcal{D}}\right] \\ &\geq \frac{1}{2}\log\left(2\pi e\sigma_n^2\right) + \frac{1}{2}(N - n)\log\left(2\pi e\sigma_n^2\right) \\ &= \frac{1}{2}(N - n + 1)\log\left(2\pi e\sigma_n^2\right) \end{aligned}$$

such that the inequality in (B.8) is due to Lemmas 11 and 12, and the last inequality in (B.8) holds because $\sigma_o \geq \sigma_n$, by definition in (4.14) (hence, $\log(\sigma_o/\sigma_n) \geq 0$). Recall that π^ϵ is a stochastic policy (instead of a deterministic policy like π^*) due to

its use of the truncated sampling procedure (Section 4.2.2), which implies $\pi_n^\epsilon(z_\mathcal{D})$ is a random variable. As a result,

$$\begin{aligned}\mathbb{E}_{\pi_n^\epsilon(z_\mathcal{D})}\left[Q_n^*(z_\mathcal{D},\pi_n^*(z_\mathcal{D}))-Q_n^*(z_\mathcal{D},\pi_n^\epsilon(z_\mathcal{D}))\right] &\leq (1-\delta)(2N\gamma)+\delta\left(N\log\left(\frac{\sigma_o}{\sigma_n}\right)+\log|\Lambda|\right)\\ &\leq 2N\gamma+\delta\left(N\log\left(\frac{\sigma_o}{\sigma_n}\right)+\log|\Lambda|\right) \qquad \text{(B.8)}\end{aligned}$$

where the expectation is with respect to random variable $\pi^\epsilon(z_\mathcal{D})$ and the first inequality follows from Lemma 4 and (B.8). Using the facts that

$$\mathbb{E}_{\pi_n^\epsilon(z_\mathcal{D})}\left[Q_n^*(z_\mathcal{D},\pi_n^*(z_\mathcal{D}))-Q_n^*(z_\mathcal{D},\pi_n^\epsilon(z_\mathcal{D}))\right] = V_n^*(z_\mathcal{D})-\mathbb{E}_{\pi_n^\epsilon(z_\mathcal{D})}\left[Q_n^*(z_\mathcal{D},\pi^\epsilon(z_\mathcal{D}))\right]$$

and $\mathbb{E}_{\pi_n^\epsilon(z_\mathcal{D})}\left[Q_n^*\left(z_\mathcal{D},\pi_n^\epsilon(z_\mathcal{D})\right)\right]=\mathbb{E}_{\pi_n^\epsilon(z_\mathcal{D})}\left[\mathbb{H}[Z_{\pi_n^\epsilon(z_\mathcal{D})}|z_\mathcal{D}]+\mathbb{E}\left[V_{n+1}^*\left(z_\mathcal{D}\cup\{Z_{\pi_n^\epsilon(z_\mathcal{D})}\}\right)|z_\mathcal{D}\right]\right]$, (B.8) therefore becomes

$$\begin{aligned}V_n^*(z_\mathcal{D})-\mathbb{E}_{\pi_n^\epsilon(z_\mathcal{D})}\left[\mathbb{H}[Z_{\pi_n^\epsilon(z_\mathcal{D})}|z_\mathcal{D}]\right] &\leq \mathbb{E}_{\pi_n^\epsilon(z_\mathcal{D})}\left[\mathbb{E}\left[V_{n+1}^*\big(z_\mathcal{D}\cup\{Z_{\pi_n^\epsilon(z_\mathcal{D})}\}\big)|z_\mathcal{D}\right]\right]\\ &\quad +2N\gamma+\delta\left(N\log\left(\frac{\sigma_o}{\sigma_n}\right)+\log|\Lambda|\right) \qquad \text{(B.9)}\end{aligned}$$

such that there is no expectation term on the RHS expression of (B.9) when $n=N$.

From (4.6), $V_1^\pi(z_{\mathcal{D}_0})$ can be expanded into the following recursive formulation using chain rule for entropy:

$$V_n^\pi(z_\mathcal{D}) = \mathbb{H}\left[Z_{\pi_n(z_\mathcal{D})}|z_\mathcal{D}\right]+\mathbb{E}\left[V_{n+1}^\pi\left(z_\mathcal{D}\cup\{Z_{\pi_n(z_\mathcal{D})}\}\right)|z_\mathcal{D}\right] \qquad \text{(B.10)}$$

for stage $n=1,\ldots,N$ where the expectation term is omitted from the RHS expression of V_N^π at stage N. Using (B.9) and (B.10) above, we will now give a proof by induction on n that

$$V_n^*(z_{\mathcal{D}}) - \mathbb{E}_{\{\pi_i^\epsilon\}_{i=n}^N}\left[V_n^{\pi^\epsilon}(z_{\mathcal{D}})\right] \;\leq\; (N-n+1)\left(2N\gamma + \delta\left(N\log\left(\frac{\sigma_o}{\sigma_n}\right) + \log|\Lambda|\right)\right). \tag{B.11}$$

When $n = N$,

$$\begin{aligned} V_N^*(z_{\mathcal{D}}) - \mathbb{E}_{\{\pi_N^\epsilon\}}\left[V_N^{\pi^\epsilon}(z_{\mathcal{D}})\right] &= V_N^*(z_{\mathcal{D}}) - \mathbb{E}_{\pi_N^\epsilon(z_{\mathcal{D}})}\left[\mathbb{H}[Z_{\pi_N^\epsilon(z_{\mathcal{D}})}|z_{\mathcal{D}}]\right] \\ &\leq 2N\gamma + \delta\left(N\log\left(\frac{\sigma_o}{\sigma_n}\right) + \log|\Lambda|\right) \end{aligned}$$

such that the equality is due to (B.10) and the inequality follows from (B.9). So, (B.11) holds for the base case. Supposing (B.11) holds for $n+1$ (i.e., induction hypothesis), we will prove that it holds for $n < N$:

$$\begin{aligned} V_n^*(z_{\mathcal{D}}) - \mathbb{E}_{\{\pi_i^\epsilon\}_{i=n}^N}\left[V_n^{\pi^\epsilon}(z_{\mathcal{D}})\right] &= V_n^*(z_{\mathcal{D}}) - \mathbb{E}_{\pi_n^\epsilon(z_{\mathcal{D}})}\left[\mathbb{H}\left[Z_{\pi_n^\epsilon(z_{\mathcal{D}})}|z_{\mathcal{D}}\right]\right] \\ &- \mathbb{E}_{\pi_n^\epsilon(z_{\mathcal{D}})}\left[\mathbb{E}\left[\mathbb{E}_{\{\pi_i^\epsilon\}_{i=n+1}^N}\left[V_{n+1}^{\pi^\epsilon}\left(z_{\mathcal{D}}\cup\{Z_{\pi_n^\epsilon(z_{\mathcal{D}})}\}\right)\right]\Big|z_{\mathcal{D}}\right]\right] \\ &\leq \mathbb{E}_{\pi_n^\epsilon(z_{\mathcal{D}})}\left[\mathbb{E}\left[V_{n+1}^*\left(z_{\mathcal{D}}\cup\{Z_{\pi_n^\epsilon(z_{\mathcal{D}})}\}\right)|z_{\mathcal{D}}\right]\right] \\ &+ 2N\gamma + \delta\left(N\log\left(\frac{\sigma_o}{\sigma_n}\right) + \log|\Lambda|\right) \\ &- \mathbb{E}_{\pi_n^\epsilon(z_{\mathcal{D}})}\left[\mathbb{E}\left[\mathbb{E}_{\{\pi_i^\epsilon\}_{i=n+1}^N}\left[V_{n+1}^{\pi^\epsilon}\left(z_{\mathcal{D}}\cup\{Z_{\pi_n^\epsilon(z_{\mathcal{D}})}\}\right)\right]\Big|z_{\mathcal{D}}\right]\right] \\ &= \mathbb{E}_{\pi_n^\epsilon(z_{\mathcal{D}})}\Big[\mathbb{E}\Big[V_{n+1}^*\left(z_{\mathcal{D}}\cup\{Z_{\pi_n^\epsilon(z_{\mathcal{D}})}\}\right) \\ &\qquad -\mathbb{E}_{\{\pi_i^\epsilon\}_{i=n+1}^N}\left[V_{n+1}^{\pi^\epsilon}\left(z_{\mathcal{D}}\cup\{Z_{\pi_n^\epsilon(z_{\mathcal{D}})}\}\right)\right]\Big|z_{\mathcal{D}}\Big]\Big] \\ &+ 2N\gamma + \delta\left(N\log\left(\frac{\sigma_o}{\sigma_n}\right) + \log|\Lambda|\right) \\ &\leq (N-n)\left(2N\gamma + \delta\left(N\log\left(\frac{\sigma_o}{\sigma_n}\right) + \log|\Lambda|\right)\right) \\ &+ 2N\gamma + \delta\left(N\log\left(\frac{\sigma_o}{\sigma_n}\right) + \log|\Lambda|\right) \\ &= (N-n+1)\left(2N\gamma + \delta\left(N\log\left(\frac{\sigma_o}{\sigma_n}\right) + \log|\Lambda|\right)\right). \end{aligned}$$

such that the first equality is due to (B.10), and the first and second inequalities follow from (B.9) and induction hypothesis, respectively.

From (B.11), when $n = 1$,

$$\begin{aligned} V_1^*(z_{\mathcal{D}}) - \mathbb{E}_{\pi^\epsilon}\left[V_1^{\pi^\epsilon}(z_{\mathcal{D}})\right] &= V_1^*(z_{\mathcal{D}}) - \mathbb{E}_{\{\pi_i^\epsilon\}_{i=1}^N}\left[V_1^{\pi^\epsilon}(z_{\mathcal{D}})\right] \\ &\leq N\left(2N\gamma + \delta\left(N\log\left(\frac{\sigma_o}{\sigma_n}\right) + \log|\Lambda|\right)\right). \end{aligned}$$

Let $\epsilon = N(2N\gamma + \delta(N\log(\sigma_o/\sigma_n) + \log|\Lambda|))$ by setting $\gamma = \epsilon/(4N^2)$ and $\delta = \epsilon/(2N(N\log(\sigma_o/\sigma_n) + \log|\Lambda|))$. As a result, from Lemma 4, $\tau = \mathcal{O}(\sqrt{\log(1/\epsilon)})$ and consequently, it follows that

$$S = \mathcal{O}\left(\frac{\left(\log\left(\frac{1}{\epsilon}\right)\right)^2}{\epsilon^2}\log\left(\frac{\log\left(\frac{1}{\epsilon}\right)}{\epsilon^2}\right)\right).$$

Theorem 6 then follows. □

B.7 Proof of Theorem 15

Theorem 15. *Let π be any stochastic policy. Then, $\mathbb{E}_\pi\left[V_1^\pi(z_{\mathcal{D}_0})\right] \leq V_1^*(z_{\mathcal{D}_0})$.*

Proof. We will give a proof by induction on n that

$$\mathbb{E}_{\{\pi_i\}_{i=n}^N}\left[V_n^\pi(z_{\mathcal{D}})\right] \leq V_n^*(z_{\mathcal{D}}) \ .$$

When $n = N$, we have

$$\begin{aligned}
\mathbb{E}_{\{\pi_N\}}\left[V_N^{\pi}\left(z_{\mathcal{D}}\right)\right] &= \mathbb{E}_{\pi_N(z_{\mathcal{D}})}\left[\mathbb{H}\left[Z_{\pi_N(z_{\mathcal{D}})}|z_{\mathcal{D}}\right]\right] \\
&\leq \mathbb{E}_{\pi_N(z_{\mathcal{D}})}\left[\max_{x\in\mathcal{X}\setminus\mathcal{D}} \mathbb{H}\left[Z_x|z_{\mathcal{D}}\right]\right] \\
&= \mathbb{E}_{\pi_N(z_{\mathcal{D}})}\left[V_N^{*}\left(z_{\mathcal{D}}\right)\right] \;=\; V_N^{*}\left(z_{\mathcal{D}}\right)
\end{aligned}$$

such that the first and second last equalities are due to (B.10) and (4.7), respectively. So, (B.12) holds for the base case. Supposing (B.12) holds for $n+1$ (i.e., induction hypothesis), we will prove that it holds for $n < N$:

$$\begin{aligned}
\mathbb{E}_{\{\pi_i\}_{i=n}^{N}}\left[V_n^{\pi}\left(z_{\mathcal{D}}\right)\right] &= \mathbb{E}_{\pi_n(z_{\mathcal{D}})}\left[\mathbb{H}\left[Z_{\pi_n(z_{\mathcal{D}})}|z_{\mathcal{D}}\right]\right] \\
&+ \mathbb{E}_{\pi_n(z_{\mathcal{D}})}\left[\mathbb{E}\left[\mathbb{E}_{\{\pi_i\}_{i=n+1}^{N}}\left[V_{n+1}^{\pi}\left(z_{\mathcal{D}}\cup\{Z_{\pi_n(z_{\mathcal{D}})}\}\right)\right]\Big|z_{\mathcal{D}}\right]\right] \\
&\leq \mathbb{E}_{\pi_n(z_{\mathcal{D}})}\left[\mathbb{H}\left[Z_{\pi_n(z_{\mathcal{D}})}|z_{\mathcal{D}}\right]\right] + \mathbb{E}_{\pi_n(z_{\mathcal{D}})}\left[\mathbb{E}\left[V_{n+1}^{*}\left(z_{\mathcal{D}}\cup\{Z_{\pi_n(z_{\mathcal{D}})}\}\right)\Big|z_{\mathcal{D}}\right]\right] \\
&\leq \mathbb{E}_{\pi_n(z_{\mathcal{D}})}\left[\max_{x\in\mathcal{X}\setminus\mathcal{D}}\left(\mathbb{H}\left[Z_x|z_{\mathcal{D}}\right] + \mathbb{E}\left[V_{n+1}^{*}\left(z_{\mathcal{D}}\cup\{Z_x\}\right)|z_{\mathcal{D}}\right]\right)\right] \\
&= \mathbb{E}_{\pi_n(z_{\mathcal{D}})}\left[V_n^{*}\left(z_{\mathcal{D}}\right)\right] = V_n^{*}\left(z_{\mathcal{D}}\right)
\end{aligned}$$

such that the first and second last equalities are, respectively, due to (B.10) and (4.7), and the first inequality follows from the induction hypothesis. □

Appendix C

Proofs of Auxiliary Results for Chapter 4

C.1 Lemma 7

Lemma 7. *For all $z_{\mathcal{D}}$, $x \in \mathcal{X} \setminus \mathcal{D}$, and $\lambda = \{\sigma_n^\lambda, \sigma_s^\lambda, \ell_1^\lambda, \ldots, \ell_P^\lambda\} \in \Lambda$ (Section 4.1), $\sigma_n^2 \leq \sigma_{xx|\mathcal{D},\lambda} \leq \sigma_o^2$ where σ_n^2 and σ_o^2 are defined in* (4.14).

Proof. Lemma 6 of Cao *et al.* [2013] implies $(\sigma_n^\lambda)^2 \leq \sigma_{xx|\mathcal{D},\lambda} \leq (\sigma_s^\lambda)^2 + (\sigma_n^\lambda)^2$, from which Lemma 7 directly follows. □

C.2 Lemma 8

Lemma 8. *Let $[-\widehat{\tau}, \widehat{\tau}]$ ($[-\widehat{\tau}', \widehat{\tau}']$) denote the support of the distribution of $\widehat{Z}_x$ ($\widehat{Z}_{x'}$) for all $x \in \mathcal{X} \setminus \mathcal{D}$ ($x' \in \mathcal{X} \setminus (\mathcal{D} \cup \{x\})$) at stage n ($n+1$) for $n = 1, \ldots, N-1$. Then,*

$$\widehat{\tau}' \leq \kappa \widehat{\tau} - \frac{\kappa - 3}{2}\tau \tag{C.1}$$

where κ is previously defined in (4.13)*. Without loss of generality, assuming $\mu_{x|\mathcal{D}_0,\lambda} = 0$ for all $x \in \mathcal{X} \setminus \mathcal{D}_0$ and $\lambda \in \Lambda$, $\widehat{\tau} \leq n\kappa^{n-1}\tau$ at stage $n = 1, \ldots, N$.*

Proof. By Definition 1, since $|\mu_{x'|\mathcal{D},\lambda}| \leq \widehat{\tau} - \tau$, $|z_x^i| \leq \widehat{\tau}$, and $|\mu_{x|\mathcal{D},\lambda}| \leq \widehat{\tau} - \tau$, it follows from (4.13) and the following property of Gaussian posterior mean

$$\mu_{x'|\mathcal{D}\cup\{x\},\lambda} \;\; = \;\; \mu_{x'|\mathcal{D},\lambda} + \sigma_{x'x|\mathcal{D},\lambda}\sigma^{-1}_{xx|\mathcal{D},\lambda}(z_x^i - \mu_{x|\mathcal{D},\lambda})$$

that $|\mu_{x'|\mathcal{D}\cup\{x\},\lambda}| \leq \kappa\widehat{\tau} - 0.5(\kappa - 1)\tau$. Consequently, $|\min_{x'\in\mathcal{X}\setminus(\mathcal{D}\cup\{x\}),\lambda\in\Lambda} \mu_{x'|\mathcal{D}\cup\{x\},\lambda} - \tau| \leq \kappa\widehat{\tau} - 0.5(\kappa - 3)\tau$ and $|\max_{x'\in\mathcal{X}\setminus(\mathcal{D}\cup\{x\}),\lambda\in\Lambda} \mu_{x'|\mathcal{D}\cup\{x\},\lambda} + \tau| \leq \kappa\widehat{\tau} - 0.5(\kappa - 3)\tau$. Then, $\widehat{\tau}' \leq \kappa\widehat{\tau} - 0.5(\kappa - 3)\tau$, by (4.8).

Since $\mu_{x|\mathcal{D}_0,\lambda} = 0$ for all $x \in \mathcal{X}\setminus\mathcal{D}_0$ and $\lambda \in \Lambda$, $\widehat{\tau} = \tau$ at stage $n = 1$, by (4.8). If $\kappa \geq 3$, then it follows from (C.1) that $\widehat{\tau}' \leq \kappa\widehat{\tau} - 0.5(\kappa - 3)\tau \leq \kappa\widehat{\tau}$ since $0 \leq 0.5(\kappa - 3) \leq \kappa$ and $0 \leq \tau \leq \widehat{\tau}$. As a result, $\widehat{\tau} \leq \kappa^{n-1}\tau$ at stage $n = 1, \ldots, N$. Otherwise (i.e., $1 \leq \kappa < 3$), $\widehat{\tau}' \leq \kappa\widehat{\tau} + 0.5(3 - \kappa)\tau \leq \kappa\widehat{\tau} + \tau$ since $0 < 0.5(3 - \kappa) \leq 1$. Consequently, $\widehat{\tau} \leq \sum_{i=0}^{n-1} \kappa^i\tau \leq n\kappa^{n-1}\tau$ at stage $n = 1, \ldots, N$. □

C.3 Lemma 9

Definition 2 (Diagonally Dominant $\Sigma_{\mathcal{D}\mathcal{D}|\lambda}$)**.** *Given $z_{\mathcal{D}}$ ($\mathcal{D} \subseteq \mathcal{X}$) and $\lambda \in \Lambda$, $\Sigma_{\mathcal{D}\mathcal{D}|\lambda}$ is said to be diagonally dominant if*

$$\sigma_{xx|\lambda} \geq \left(\sqrt{|\mathcal{D}| - 1} + 1\right) \sum_{x'\in\mathcal{D}\setminus\{x\}} \sigma_{xx'|\lambda}$$

for any $x \in \mathcal{D}$. *Furthermore, since* $\sigma_{xx|\lambda} = \left(\sigma_s^\lambda\right)^2 + \left(\sigma_n^\lambda\right)^2$ *for all* $x \in \mathcal{X}$,

$$\sigma_{xx|\lambda} \geq \left(\sqrt{|\mathcal{D}|-1}+1\right) \max_{u \in \mathcal{D}} \sum_{x' \in \mathcal{D}\setminus\{u\}} \sigma_{ux'|\lambda} \ .$$

Lemma 9. *Without loss of generality, assume that* $\mu_x = 0$ *for all* $x \in \mathcal{X}$. *For all* $z_\mathcal{D}$ *(*$\mathcal{D} \subseteq \mathcal{X}$*),* $\lambda \in \Lambda$, *and* $\eta > 0$, *if* $\Sigma_{\mathcal{D}\mathcal{D}|\lambda}$ *is diagonally dominant (Definition 2) and* $|z_u| \leq \eta$ *for all* $u \in \mathcal{D}$, *then* $|\mu_{x|\mathcal{D},\lambda}| \leq \eta$ *for all* $x \in \mathcal{X} \setminus \mathcal{D}$.

Proof. Since $\mu_x = 0$ for all $x \in \mathcal{X}$,

$$\mu_{x|\mathcal{D},\lambda} \quad = \quad \Sigma_{x\mathcal{D}|\lambda}\Sigma^{-1}_{\mathcal{D}\mathcal{D}|\lambda} z_\mathcal{D} \ . \tag{C.2}$$

Since $\Sigma^{-1}_{\mathcal{D}\mathcal{D}|\lambda}$ is a symmetric, positive-definite matrix, there exists an orthonormal basis comprising the eigenvectors $E \triangleq [e_1 \ e_2 \ \ldots \ e_{|\mathcal{D}|}]$ ($e_i^\top e_i = 1$ and $e_i^\top e_j = 0$ for $i \neq j$) and their associated positive eigenvalues $\Psi^{-1} \triangleq \text{Diag}[\psi_1^{-1}, \psi_2^{-1}, \ldots, \psi_{|\mathcal{D}|}^{-1}]$ such that $\Sigma^{-1}_{\mathcal{D}\mathcal{D}|\lambda} = E\Psi^{-1}E^\top$ (i.e., spectral theorem). Denote $\{\alpha_i\}_{i=1}^{|\mathcal{D}|}$ and $\{\beta_i\}_{i=1}^{|\mathcal{D}|}$ as the sets of coefficients when $\Sigma_{\mathcal{D}x|\lambda}$ and $z_\mathcal{D}$ are projected on E, respectively. (C.2) can therefore be rewritten as

$$\begin{aligned}
\mu_{x|\mathcal{D},\lambda} &= \left(\sum_{i=1}^{|\mathcal{D}|} \alpha_i e_i^\top\right) \Sigma^{-1}_{\mathcal{D}\mathcal{D}|\lambda} \left(\sum_{i=1}^{|\mathcal{D}|} \beta_i e_i\right) \\
&= \left(\sum_{i=1}^{|\mathcal{D}|} \alpha_i e_i^\top\right) \left(\sum_{i=1}^{|\mathcal{D}|} \beta_i \left(\Sigma^{-1}_{\mathcal{D}\mathcal{D}|\lambda} e_i\right)\right) \\
&= \left(\sum_{i=1}^{|\mathcal{D}|} \alpha_i e_i^\top\right) \left(\sum_{i=1}^{|\mathcal{D}|} \beta_i \psi_i^{-1} e_i\right) \\
&= \sum_{i=1}^{|\mathcal{D}|} \alpha_i \beta_i \psi_i^{-1} \ .
\end{aligned} \tag{C.3}$$

From (C.3), $\mu_{x|\mathcal{D},\lambda}^2 = \left(\sum_{i=1}^{|\mathcal{D}|} \alpha_i \beta_i \psi_i^{-1}\right)^2 \leq \psi_{\min}^{-2} \left(\sum_{i=1}^{|\mathcal{D}|} \alpha_i^2\right) \left(\sum_{i=1}^{|\mathcal{D}|} \beta_i^2\right) = \psi_{\min}^{-2} \left\|\Sigma_{x\mathcal{D}|\lambda}\right\|_2^2 \|z_{\mathcal{D}}\|_2^2$ with $\psi_{\min} \triangleq \min_{i=1}^{|\mathcal{D}|} \psi_i$, which can be bounded from below by applying Gershgorin circle theorem for $\Sigma_{\mathcal{D}\mathcal{D}|\lambda}$:

$$\begin{aligned}
\psi_{\min} &\geq \min_{u\in\mathcal{D}} \left(\sigma_{uu|\lambda} - R_{\mathcal{D}}^{\lambda}(u)\right) \\
&= \sigma_{xx|\lambda} - \max_{u\in\mathcal{D}} R_{\mathcal{D}}^{\lambda}(u) \\
&\geq \left(\sqrt{|\mathcal{D}|}+1\right) \max_{u\in\mathcal{D}\cup\{x\}} R_{\mathcal{D}\cup\{x\}}^{\lambda}(u) - \max_{u\in\mathcal{D}} R_{\mathcal{D}}^{\lambda}(u)
\end{aligned}$$

where $R_{\mathcal{D}}^{\lambda}(u) \triangleq \sum_{x'\in\mathcal{D}\setminus\{u\}} \sigma_{ux'|\lambda}$, the first equality follows from the fact that $\sigma_{uu|\lambda} = \left(\sigma_s^{\lambda}\right)^2 + \left(\sigma_n^{\lambda}\right)^2 = \sigma_{xx|\lambda}$ for all $u, x \in \mathcal{X}$, and the second inequality holds because $\Sigma_{(\mathcal{D}\cup\{x\})(\mathcal{D}\cup\{x\})|\lambda}$ is assumed to be diagonally dominant (Definition 2). On the other hand, since $x \notin \mathcal{D}$, $R_{\mathcal{D}\cup\{x\}}^{\lambda}(u) = R_{\mathcal{D}}^{\lambda}(u) + \sigma_{ux|\lambda} \geq R_{\mathcal{D}}^{\lambda}(u)$ for all $u \in \mathcal{D}$, which immediately implies $\max_{u\in\mathcal{D}\cup\{x\}} R_{\mathcal{D}\cup\{x\}}^{\lambda}(u) \geq \max_{u\in\mathcal{D}} R_{\mathcal{D}\cup\{x\}}^{\lambda}(u) \geq \max_{u\in\mathcal{D}} R_{\mathcal{D}}^{\lambda}(u)$. Plugging this into the above inequality, $\psi_{\min} \geq \left(\sqrt{|\mathcal{D}|}+1\right) \max_{u\in\mathcal{D}\cup\{x\}} R_{\mathcal{D}\cup\{x\}}^{\lambda}(u) - \max_{u\in\mathcal{D}} R_{\mathcal{D}}^{\lambda}(u) \geq \sqrt{|\mathcal{D}|} \max_{u\in\mathcal{D}\cup\{x\}} R_{\mathcal{D}\cup\{x\}}^{\lambda}(u) \geq \sqrt{|\mathcal{D}|} R_{\mathcal{D}\cup\{x\}}^{\lambda}(x)$. Since $\|\Sigma_{x\mathcal{D}|\lambda}\|_2 = \sqrt{\sum_{u\in\mathcal{D}} \sigma_{xu|\lambda}^2} \leq \sum_{u\in\mathcal{D}} \sigma_{xu|\lambda} = R_{\mathcal{D}\cup\{x\}}^{\lambda}(x)$, it follows that $\psi_{\min} \geq \sqrt{|\mathcal{D}|}\|\Sigma_{x\mathcal{D}|\lambda}\|_2$ or, equivalently, $\psi_{\min}^2 \geq |\mathcal{D}|\|\Sigma_{x\mathcal{D}|\lambda}\|_2^2$, which implies $\mu_{x|\mathcal{D},\lambda}^2 \leq \psi_{\min}^{-2}\|\Sigma_{x\mathcal{D}|\lambda}\|_2^2\|z_{\mathcal{D}}\|_2^2 \leq |\mathcal{D}|^{-1}\|z_{\mathcal{D}}\|_2^2 \leq |\mathcal{D}|^{-1}|\mathcal{D}|\eta^2 = \eta^2$ where the last inequality holds due to the fact that $|z_u| \leq \eta$ for all $u \in \mathcal{D}$. Hence, $|\mu_{x|\mathcal{D},\lambda}| \leq \eta$. □

C.4 Lemma 10

Lemma 10. *Let $[-\widehat{\tau}_{\max}, \widehat{\tau}_{\max}]$ and $[-\widehat{\tau}, \widehat{\tau}]$ denote the largest support of the distributions of $\widehat{Z}_x$ for all $x \in \mathcal{X} \setminus \mathcal{D}$ at stages $1, 2, \ldots, n$ and the support of the distribution of $\widehat{Z}_x$ for all $x \in \mathcal{X} \setminus \mathcal{D}$ at stage $n+1$ for $n = 1, 2, \ldots, N-1$, respectively. Suppose that $\mathcal{D}_0 = \emptyset$ and, without loss of generality, $\mu_x = 0$ for all $x \in \mathcal{X}$. For all $z_{\mathcal{D}}$ ($\mathcal{D} \subseteq \mathcal{X}$) and $\lambda \in \Lambda$, if $\Sigma_{\mathcal{D}\mathcal{D}|\lambda}$ is diagonally dominant (Definition 2), then $\widehat{\tau} \leq \widehat{\tau}_{\max} + \tau$. Con-*

sequently, $\widehat{\tau} \leq n\tau$ *at stage* $n = 1, \ldots, N$.

Remark. If $\Sigma_{\mathcal{D}\mathcal{D}|\lambda}$ is diagonally dominant (Definition 2), then Lemma 10 provides a tighter bound on $\widehat{\tau}$ (i.e., $\widehat{\tau} \leq n\tau$) than Lemma 8 that does not involve κ. In fact, it coincides exactly with the bound derived in Lemma 8 by setting $\kappa = 1$. By using this bound (instead of Lemma 8's bound) in the proof of Lemma 2 (Appendix B.2), it is easy to see that the probabilistic bound in Lemma 2 and its subsequent results hold by setting $\kappa = 1$.

Proof. Since $[-\widehat{\tau}_{\max}, \widehat{\tau}_{\max}]$ is the largest support of the distributions of $\widehat{Z}_x$ for all $x \in \mathcal{X} \setminus \mathcal{D}$ at stages $1, 2, \ldots, n$, $|z_x^i| \leq \widehat{\tau}_{\max}$ for all $x \in \mathcal{X} \setminus \mathcal{D}$ at stages $1, 2, \ldots, n$, by Definition 1. Therefore, at stage $n+1$, $|z_u| \leq \widehat{\tau}_{\max}$ for all $u \in \mathcal{D}$. By Lemma 9, $|\mu_{x|\mathcal{D},\lambda}| \leq \widehat{\tau}_{\max}$ for all $x \in \mathcal{X} \setminus \mathcal{D}$ and $\lambda \in \Lambda$ at stage $n+1$, which consequently implies $|\min_{x \in \mathcal{X}\setminus\mathcal{D}, \lambda \in \Lambda} \mu_{x|\mathcal{D},\lambda} - \tau| \leq \widehat{\tau}_{\max} + \tau$ and $|\max_{x \in \mathcal{X}\setminus\mathcal{D}, \lambda \in \Lambda} \mu_{x|\mathcal{D},\lambda} + \tau| \leq \widehat{\tau}_{\max} + \tau$. Then, it follows from (4.8) that $\widehat{\tau} \leq \widehat{\tau}_{\max} + \tau$ at stage $n+1$ for $n = 1, \ldots, N-1$. Since $\mathcal{D}_0 = \emptyset$, $\mu_{x|\mathcal{D}_0,\lambda} = \mu_x = 0$. Then, $\widehat{\tau} = \tau$ at stage 1, by (4.8). Consequently, $\widehat{\tau} \leq n\tau$ at stage $n = 1, 2, \ldots, N$. □

C.5 Lemma 11

Lemma 11. *For all* $z_{\mathcal{D}}$ *and* $n = 1, \ldots, N$,

$$\begin{aligned} V_n^*(z_{\mathcal{D}}) &\leq \frac{1}{2}(N-n+1)\log\left(2\pi e\sigma_o^2\right) + \log|\Lambda| \,, \\ V_n^*(z_{\mathcal{D}}) &\geq \frac{1}{2}(N-n+1)\log\left(2\pi e\sigma_n^2\right) \end{aligned}$$

where σ_n^2 *and* σ_o^2 *are previously defined in* (4.14).

Proof. By definition (4.7), $V_n^*(z_{\mathcal{D}}) = \mathbb{H}\left[Z_{\{\pi_i^*\}_{i=n}^N} | z_{\mathcal{D}}\right]$. Using Theorem 1 of Krause

and Guestrin [2007],

$$\begin{aligned} \mathbb{H}\left[Z_{\{\pi_i^*\}_{i=n}^N}|z_{\mathcal{D}}\right] &\leq \sum_{\lambda\in\Lambda} b_{\mathcal{D}}(\lambda) \max_{|\mathcal{A}|=N-n+1} \mathbb{H}\left[Z_{\mathcal{A}}|z_{\mathcal{D}},\lambda\right] + \mathbb{H}\left[\Lambda\right] \\ &= \sum_{\lambda\in\Lambda} b_{\mathcal{D}}(\lambda)\ \mathbb{H}\left[Z_{\mathcal{A}^\lambda}|z_{\mathcal{D}},\lambda\right] + \mathbb{H}\left[\Lambda\right] \end{aligned} \tag{C.4}$$

where Λ denotes the set of random parameters corresponding to the realized parameters λ, $\mathcal{A}, \mathcal{A}^\lambda \subseteq \mathcal{X} \setminus \mathcal{D}$, $\mathcal{A}^\lambda \triangleq \arg\max_{|\mathcal{A}|=N-n+1} \mathbb{H}\left[Z_{\mathcal{A}}|z_{\mathcal{D}},\lambda\right]$, and

$$\begin{aligned} \mathbb{H}[Z_{\mathcal{A}}|z_{\mathcal{D}},\lambda] &\triangleq -\int p(z_{\mathcal{A}}|z_{\mathcal{D}},\lambda)\log p(z_{\mathcal{A}}|z_{\mathcal{D}},\lambda)\ \mathrm{d}z_{\mathcal{A}} \\ &= \frac{1}{2}\log\left((2\pi e)^{|\mathcal{A}|}\left|\Sigma_{\mathcal{A}\mathcal{A}|\mathcal{D},\lambda}\right|\right) \end{aligned} \tag{C.5}$$

such that $\Sigma_{\mathcal{A}\mathcal{A}|\mathcal{D},\lambda}$ is a posterior covariance matrix with components $\sigma_{xx'|\mathcal{D},\lambda}$ for all $x, x' \in \mathcal{A}$. Furthermore, we have

$$\begin{aligned} \mathbb{H}[Z_{\mathcal{A}^\lambda}|z_{\mathcal{D}},\lambda] &\leq \sum_{x\in\mathcal{A}^\lambda} \mathbb{H}[Z_x|z_{\mathcal{D}},\lambda] \\ &= \frac{1}{2}\sum_{x\in\mathcal{A}^\lambda} \log\left(2\pi e\sigma_{xx|\mathcal{D},\lambda}\right) \\ &\leq \frac{|\mathcal{A}^\lambda|}{2}\log\left(2\pi e\sigma_o^2\right) \\ &= \frac{1}{2}(N-n+1)\log\left(2\pi e\sigma_o^2\right) \end{aligned} \tag{C.6}$$

where $\mathbb{H}[Z_x|z_{\mathcal{D}},\lambda]$ is defined in a similar manner as (C.5). Substituting (C.6) back into (C.4), we have

$$\begin{aligned} V_n^*(z_{\mathcal{D}}) &= \mathbb{H}[Z_{\{\pi_i^*\}_{i=n}^N}|z_{\mathcal{D}}] \\ &\leq \frac{1}{2}(N-n+1)\log\left(2\pi e\sigma_o^2\right) + \mathbb{H}[\Lambda] \\ &\leq \frac{1}{2}(N-n+1)\log\left(2\pi e\sigma_o^2\right) + \log|\Lambda| \end{aligned}$$

where the last inequality follows from the fact that the entropy of a discrete distri-

bution is maximized when the distribution is uniform.

On the other hand, from (4.7),

$$\begin{aligned} V_n^*(z_{\mathcal{D}}) &= \mathbb{H}\left[Z_{\pi_n^*(z_{\mathcal{D}})}|z_{\mathcal{D}}\right] + \mathbb{E}\left[V_{n+1}^*\left(z_{\mathcal{D}} \cup \{Z_{\pi_n^*(z_{\mathcal{D}})}\}\right)\middle| z_{\mathcal{D}}\right] \\ &\geq \frac{1}{2}\log\left(2\pi e\sigma_n^2\right) + \mathbb{E}\left[V_{n+1}^*\left(z_{\mathcal{D}} \cup \{Z_{\pi_n^*(z_{\mathcal{D}})}\}\right)\middle| z_{\mathcal{D}}\right] \end{aligned} \tag{C.7}$$

where the inequality is due to Lemma 12. Then, the lower bound of $V_n^*(z_{\mathcal{D}})$ can be proven by induction using (C.7), as detailed next. When $n = N$ (i.e., base case), $V_N^*(z_{\mathcal{D}}) = \mathbb{H}[Z_{\pi_N^*(z_{\mathcal{D}})}|z_{\mathcal{D}}] \geq 0.5\log\left(2\pi e\sigma_n^2\right)$, by Lemma 12. Supposing $V_{n+1}^*(z_{\mathcal{D}}) \geq 0.5(N-n)\log\left(2\pi e\sigma_n^2\right)$ for $n < N$ (i.e., induction hypothesis), $V_n^*(z_{\mathcal{D}}) \geq 0.5(N-n+1)\log\left(2\pi e\sigma_n^2\right)$, by (C.7). □

C.6 Lemma 12

Lemma 12. *For all $z_{\mathcal{D}}$ and $x \in \mathcal{X} \setminus \mathcal{D}$,*

$$\mathbb{H}\left[Z_x|z_{\mathcal{D}}\right] \geq \frac{1}{2}\log\left(2\pi e\sigma_n^2\right).$$

where σ_n^2 is previously defined in (4.14).

Proof. Using the monotonicity of conditional entropy (i.e., "information never hurts"

bound) [Cover and Thomas, 1991],

$$\begin{aligned}\mathbb{H}\left[Z_x|z_{\mathcal{D}}\right] &\geq \sum_{\lambda\in\Lambda} b_{\mathcal{D}}(\lambda)\ \mathbb{H}[Z_x|z_{\mathcal{D}},\lambda] \\ &= \frac{1}{2}\sum_{\lambda\in\Lambda} b_{\mathcal{D}}(\lambda)\log\left(2\pi e\sigma_{xx|\mathcal{D},\lambda}\right) \\ &\geq \frac{1}{2}\sum_{\lambda\in\Lambda} b_{\mathcal{D}}(\lambda)\log\left(2\pi e\sigma_n^2\right) \\ &= \frac{1}{2}\log\left(2\pi e\sigma_n^2\right)\end{aligned}$$

where $\mathbb{H}[Z_x|z_{\mathcal{D}},\lambda]$ is defined in a similar manner as (C.5) and the last inequality holds due to Lemma 7. □

C.7 Lemma 13

Lemma 13. *For all $z_{\mathcal{D}}$ and $x \in \mathcal{X}\setminus\mathcal{D}$,*

$$\int_{|z_x|\geq\widehat{\tau}} p(z_x|z_{\mathcal{D}})\ \mathrm{d}z_x \ \leq\ 2\Phi\left(-\frac{\tau}{\sigma_o}\right)$$

where Φ denotes the cumulative distribution function of $\mathcal{N}(0,1)$ and σ_o is previously defined in (4.14).

Proof. From (4.5),

$$\begin{aligned}
\int_{|z_x|\geq\widehat{\tau}} p(z_x|z_{\mathcal{D}})\ \mathrm{d}z_x &= \sum_{\lambda\in\Lambda} b_{\mathcal{D}}(\lambda) \int_{|z_x|\geq\widehat{\tau}} p(z_x|z_{\mathcal{D}},\lambda)\ \mathrm{d}z_x \\
&= \sum_{\lambda\in\Lambda} b_{\mathcal{D}}(\lambda) \int_{|y_x+\mu_{x|\mathcal{D},\lambda}|\geq\widehat{\tau}} p(y_x|z_{\mathcal{D}},\lambda)\ \mathrm{d}y_x \\
&\leq \sum_{\lambda\in\Lambda} b_{\mathcal{D}}(\lambda) \int_{|y_x|\geq\tau} p(y_x|z_{\mathcal{D}},\lambda)\ \mathrm{d}y_x \\
&= 2\sum_{\lambda\in\Lambda} b_{\mathcal{D}}(\lambda)\ \Phi\left(-\frac{\tau}{\sqrt{\sigma_{xx|\mathcal{D},\lambda}}}\right) \\
&\leq 2\Phi\left(-\frac{\tau}{\sigma_o}\right)
\end{aligned}$$

where, in the second equality, $y_x \triangleq z_x - \mu_{x|\mathcal{D},\lambda}$ and hence $p(y_x|z_{\mathcal{D}},\lambda) \sim \mathcal{N}(0,\sigma_{xx|\mathcal{D},\lambda})$, the first inequality follows from $\{y_x|\ |y_x+\mu_{x|\mathcal{D},\lambda}|\geq\widehat{\tau}\} \subseteq \{y_x|\ |y_x|\geq\tau\}$ since $|\mu_{x|\mathcal{D},\lambda}| \leq \widehat{\tau}-\tau$ due to (4.8), the last equality is due to the identity $\int_{|y|\geq\tau} p(y)\ \mathrm{d}y = 2\Phi(-\tau/\sigma)$ such that $p(y)\sim\mathcal{N}(0,\sigma^2)$, and the last inequality follows from the fact that Φ is an increasing function and $\sigma_{xx|\mathcal{D},\lambda} \leq \sigma_o^2$ due to Lemma 7. □

C.8 Lemma 14

Lemma 14. *We have*

$$\int_{|y-\mu|\geq\tau} y^2\ p(y)\ \mathrm{d}y = 2\left(\sigma^2+\mu^2\right)\Phi\left(-\frac{\tau}{\sigma}\right) + \sigma\tau\sqrt{\frac{2}{\pi}}\exp\left(-\frac{\tau^2}{2\sigma^2}\right) \quad \text{(C.8)}$$

where $p(y)\sim\mathcal{N}(\mu,\sigma^2)$ and Φ denotes the cumulative distribution function of $\mathcal{N}(0,1)$.

Proof. Consider $p(x) \sim \mathcal{N}(0, \sigma^2)$. Then,

$$\begin{aligned} \int_{|x|\geq\tau} x^2 \; p(x) \; \mathrm{d}x &= \sigma^2 - \int_{-\tau}^{\tau} x^2 \; p(x) \; \mathrm{d}x \\ &= \sigma^2 - \frac{2\sigma^2}{\sqrt{\pi}} \int_{-\frac{\tau}{\sqrt{2}\sigma}}^{\frac{\tau}{\sqrt{2}\sigma}} z^2 e^{-z^2} \mathrm{d}z \end{aligned} \tag{C.9}$$

where the last equality follows by setting $z \triangleq x/(\sqrt{2}\sigma)$. Then, using the following well-known identity:

$$\int_a^b z^2 e^{-z^2} \mathrm{d}z = \frac{1}{4}\left(\sqrt{\pi}\;\mathrm{erf}(z) - 2ze^{-z^2}\right)\Big|_a^b$$

for the second term on the RHS expression of (C.9),

$$\begin{aligned} \int_{-\tau}^{\tau} x^2 \; p(x) \; \mathrm{d}x &= \frac{\sigma^2}{2}\left(\mathrm{erf}\left(\frac{\tau}{\sqrt{2}\sigma}\right) - \mathrm{erf}\left(\frac{-\tau}{\sqrt{2}\sigma}\right)\right) - \sigma\tau\sqrt{\frac{2}{\pi}}\exp\left(-\frac{\tau^2}{2\sigma^2}\right) \\ &= \sigma^2\left(\Phi\left(\frac{\tau}{\sigma}\right) - \Phi\left(-\frac{\tau}{\sigma}\right)\right) - \sigma\tau\sqrt{\frac{2}{\pi}}\exp\left(-\frac{\tau^2}{2\sigma^2}\right) \end{aligned} \tag{C.10}$$

where the last equality follows from the identity $\Phi(z) = 0.5(1 + \mathrm{erf}(z/\sqrt{2}))$. Then, plugging (C.10) into (C.9) and using the identity $1 - \Phi(z) = \Phi(-z)$,

$$\int_{|x|\geq\tau} x^2 \; p(x) \; \mathrm{d}x = 2\sigma^2\Phi\left(-\frac{\tau}{\sigma}\right) + \sigma\tau\sqrt{\frac{2}{\pi}}\exp\left(-\frac{\tau^2}{2\sigma^2}\right)$$

Let $x \triangleq y - \mu$. Then,

$$\int_{|y-\mu|\geq\tau} y^2 \; p(y) \; \mathrm{d}y = \int_{|x|\geq\tau} x^2 \; p(x) \; \mathrm{d}x \; + 2\mu\int_{|x|\geq\tau} x \; p(x) \; \mathrm{d}x + \mu^2\int_{|x|\geq\tau} p(x) \; \mathrm{d}x$$

Finally, using the identities

$$\int_{|x|\geq\tau} x\; p(x)\; \mathrm{d}x \;=\; 0 \;\text{ and }\; \int_{|x|\geq\tau} p(x)\; \mathrm{d}x \;=\; 2\Phi\left(-\frac{\tau}{\sigma}\right)\;,$$

Lemma 14 directly follows. □

C.9 Lemma 15

Lemma 15. *Let us define*

$$G(z_{\mathcal{D}}, x, \lambda, \lambda') \;\triangleq\; \int_{|z_x|\geq\widehat{\tau}} \frac{(z_x - \mu_{x|\mathcal{D},\lambda})^2}{2\sigma_{xx|\mathcal{D},\lambda}} p(z_x|z_{\mathcal{D}}, \lambda')\; \mathrm{d}z_x\;. \tag{C.11}$$

For all $z_{\mathcal{D}}$, $x \in \mathcal{X}\setminus\mathcal{D}$, $\tau \geq 1$, and $\lambda, \lambda' \in \Lambda$,

$$G(z_{\mathcal{D}}, x, \lambda, \lambda') \;\leq\; \mathcal{O}\left(\frac{\sigma_o}{\sigma_n^2}\left(N^2\kappa^{2N}\tau + \sigma_o^2\right)\exp\!\left(-\frac{\tau^2}{2\sigma_o^2}\right)\right)$$

where σ_n and σ_o are defined in (4.14).

Proof. Let $y_x \triangleq z_x - \mu_{x|\mathcal{D},\lambda}$ and $\mu_{x|\lambda,\lambda'} \triangleq \mu_{x|\mathcal{D},\lambda'} - \mu_{x|\mathcal{D},\lambda}$. Then,

$$\begin{aligned} G(z_{\mathcal{D}}, x, \lambda, \lambda') \;&=\; \frac{1}{2\sigma_{xx|\mathcal{D},\lambda}} \int_{|y_x+\mu_{x|\mathcal{D},\lambda}|\geq\widehat{\tau}} y_x^2\; p(y_x|z_{\mathcal{D}}, \lambda')\; \mathrm{d}y_x \\ &\leq\; \frac{1}{2\sigma_{xx|\mathcal{D},\lambda}} \int_{|y_x-\mu_{x|\lambda,\lambda'}|\geq\tau} y_x^2\; p(y_x|z_{\mathcal{D}}, \lambda')\; \mathrm{d}y_x \end{aligned} \tag{C.12}$$

where $p(y_x|z_{\mathcal{D}}, \lambda') \sim \mathcal{N}(\mu_{x|\lambda,\lambda'}, \sigma_{xx|\mathcal{D},\lambda'})$, and the inequality follows from $\{y_x|\; |y_x + \mu_{x|\mathcal{D},\lambda}| \geq \widehat{\tau}\} \subseteq \{y_x|\; |y_x - \mu_{x|\lambda,\lambda'}| \geq \tau\}$ since $|\mu_{x|\mathcal{D},\lambda'}| \leq \widehat{\tau} - \tau$ due to (4.8).

Applying Lemma 14 to (C.12),

$$\begin{aligned} G(z_{\mathcal{D}}, x, \lambda, \lambda') &\leq \left(\frac{\sigma_{xx|\mathcal{D},\lambda'} + \mu^2_{x|\lambda,\lambda'}}{\sigma_{xx|\mathcal{D},\lambda}}\right) \Phi\left(-\frac{\tau}{\sqrt{\sigma_{xx|\mathcal{D},\lambda'}}}\right) + \frac{\tau\sqrt{\sigma_{xx|\mathcal{D},\lambda'}}}{\sigma_{xx|\mathcal{D},\lambda}\sqrt{2\pi}} \exp\left(-\frac{\tau^2}{2\sigma_{xx|\mathcal{D},\lambda'}}\right) \\ &\leq \left(\frac{\sigma_o^2 + 4N^2\kappa^{2N}\tau^2}{\sigma_n^2}\right) \Phi\left(-\frac{\tau}{\sigma_o}\right) + \frac{\tau\sigma_o}{\sigma_n^2\sqrt{2\pi}} \exp\left(-\frac{\tau^2}{2\sigma_o^2}\right) \end{aligned}$$

where the last inequality holds due to $\sigma_{xx|\mathcal{D},\lambda'} \leq \sigma_o^2$ and $\sigma_{xx|\mathcal{D},\lambda} \geq \sigma_n^2$, as proven in Lemma 7, and $\mu_{x|\lambda,\lambda'} = \mu_{x|\mathcal{D},\lambda'} - \mu_{x|\mathcal{D},\lambda} \leq 2\widehat{\tau} - 2\tau \leq 2N\kappa^{N-1}\tau$ by $|\mu_{x|\mathcal{D},\lambda}| \leq \widehat{\tau} - \tau$ and $|\mu_{x|\mathcal{D},\lambda'}| \leq \widehat{\tau} - \tau$) derived from (4.8) and by Lemma 8.

Finally, by applying the following Gaussian tail inequality:

$$\begin{aligned} \Phi\left(-\frac{\tau}{\sigma_o}\right) &= 1 - \Phi\left(\frac{\tau}{\sigma_o}\right) \\ &\leq \frac{\sigma_o}{\tau} \exp\left(-\frac{\tau^2}{2\sigma_o^2}\right) , \end{aligned} \tag{C.13}$$

it directly follows that

$$G(z_{\mathcal{D}}, x, \lambda, \lambda') \leq \mathcal{O}\left(\frac{\sigma_o}{\sigma_n^2}\left(N^2\kappa^{2N}\tau + \sigma_o^2\right) \exp\left(-\frac{\tau^2}{2\sigma_o^2}\right)\right)$$

since $\tau \geq 1$. □

C.10 Lemma 16

Lemma 16. *For all $z_{\mathcal{D}}$, $x \in \mathcal{X} \setminus \mathcal{D}$, and $\tau \geq 1$,*

$$0 \leq \mathbb{H}[Z_x|z_{\mathcal{D}}] - \widehat{\mathbb{H}}[\widehat{Z}_x|z_{\mathcal{D}}] \leq \mathcal{O}\left(\frac{\sigma_o}{\sigma_n^2}\left(N^2\kappa^{2N}\tau + \sigma_o^2\right) \exp\left(-\frac{\tau^2}{2\sigma_o^2}\right)\right) \tag{C.14}$$

where σ_n and σ_o are defined in (4.14)*. So,*

$$\left|\mathbb{H}\left[Z_x|z_{\mathcal{D}}\right] - \widehat{\mathbb{H}}\left[\widehat{Z}_x|z_{\mathcal{D}}\right]\right| \;\leq\; \mathcal{O}\left(\frac{\sigma_o}{\sigma_n^2}\left(N^2\kappa^{2N}\tau + \sigma_o^2\right)\exp\left(-\frac{\tau^2}{2\sigma_o^2}\right)\right). \quad \text{(C.15)}$$

Proof. From (4.7) and (B.4),

$$\begin{aligned}\mathbb{H}\left[Z_x|z_{\mathcal{D}}\right] - \widehat{\mathbb{H}}\left[\widehat{Z}_x|z_{\mathcal{D}}\right] &= \int_{-\infty}^{-\widehat{\tau}} p(z_x|z_{\mathcal{D}})\log\left(\frac{p(-\widehat{\tau}|z_{\mathcal{D}})}{p(z_x|z_{\mathcal{D}})}\right)\mathrm{d}z_x \\ &+ \int_{\widehat{\tau}}^{\infty} p(z_x|z_{\mathcal{D}})\log\left(\frac{p(\widehat{\tau}|z_{\mathcal{D}})}{p(z_x|z_{\mathcal{D}})}\right)\mathrm{d}z_x\,. \quad \text{(C.16)}\end{aligned}$$

Since $p(z_x|z_{\mathcal{D}})$ is the predictive distribution representing a mixture of Gaussian predictive distributions (4.5) whose posterior means (4.2) fall within the interval $[-\widehat{\tau},\widehat{\tau}]$ due to (4.8), it is clear that $p(-\widehat{\tau}|z_{\mathcal{D}}) \geq p(z_x|z_{\mathcal{D}})$ for all $z_x \leq -\widehat{\tau}$ and $p(\widehat{\tau}|z_{\mathcal{D}}) \geq p(z_x|z_{\mathcal{D}})$ for all $z_x \geq \widehat{\tau}$. As a result, the RHS expression of (C.16) is non-negative, that is, $\mathbb{H}[Z_x|z_{\mathcal{D}}] - \widehat{\mathbb{H}}[\widehat{Z}_x|z_{\mathcal{D}}] \geq 0$.

On the other hand, from (4.5),

$$\begin{aligned}p(z_x|z_{\mathcal{D}}) &= \sum_{\lambda\in\Lambda}\frac{1}{\sqrt{2\pi\sigma_{xx|\mathcal{D},\lambda}}}\exp\left(\frac{-(z_x-\mu_{x|\mathcal{D},\lambda})^2}{2\sigma_{xx|\mathcal{D},\lambda}}\right)b_{\mathcal{D}}(\lambda) \\ &\leq \sum_{\lambda\in\Lambda}\frac{1}{\sqrt{2\pi\sigma_{xx|\mathcal{D},\lambda}}}\,b_{\mathcal{D}}(\lambda) \\ &\leq \sum_{\lambda\in\Lambda}\frac{1}{\sigma_n\sqrt{2\pi}}\,b_{\mathcal{D}}(\lambda) = \frac{1}{\sigma_n\sqrt{2\pi}}\end{aligned}$$

such that the last inequality follows from Lemma 7. By taking log of both sides of the above inequality and setting $z_x = -\widehat{\tau}$ ($z_x = \widehat{\tau}$), $\log p(-\widehat{\tau}|z_{\mathcal{D}}) \leq -0.5\log(2\pi\sigma_n^2)$

$(\log p(\widehat{\tau}|z_{\mathcal{D}}) \leq -0.5\log(2\pi\sigma_n^2))$. Then, from (C.16),

$$\begin{aligned}\mathbb{H}\left[Z_x|z_{\mathcal{D}}\right] - \widehat{\mathbb{H}}\left[\widehat{Z}_x|z_{\mathcal{D}}\right] &\leq -\frac{1}{2}\log\left(2\pi\sigma_n^2\right)\int_{|z_x|\geq\widehat{\tau}} p(z_x|z_{\mathcal{D}})\mathrm{d}z_x \\ &+ \int_{|z_x|\geq\widehat{\tau}} p(z_x|z_{\mathcal{D}})\left(-\log p(z_x|z_{\mathcal{D}})\right)\mathrm{d}z_x\ . \qquad \text{(C.17)}\end{aligned}$$

Using (4.5) and Jensen's inequality, since $-\log$ is a convex function,

$$\begin{aligned}\int_{|z_x|\geq\widehat{\tau}} p(z_x|z_{\mathcal{D}})\left(-\log p(z_x|z_{\mathcal{D}})\right)\mathrm{d}z_x &\leq \sum_{\lambda\in\Lambda} b_{\mathcal{D}}(\lambda)\int_{|z_x|\geq\widehat{\tau}} p(z_x|z_{\mathcal{D}})\left(-\log p(z_x|z_{\mathcal{D}},\lambda)\right)\mathrm{d}z_x \\ &\leq \frac{1}{2}\log\left(2\pi\sigma_o^2\right)\int_{|z_x|\geq\widehat{\tau}} p(z_x|z_{\mathcal{D}})\mathrm{d}z_x \\ &+ \sum_{\lambda,\lambda'\in\Lambda} b_{\mathcal{D}}(\lambda)b_{\mathcal{D}}(\lambda')G(z_{\mathcal{D}},x,\lambda,\lambda') \\ &\leq \frac{1}{2}\log\left(2\pi\sigma_o^2\right)\int_{|z_x|\geq\widehat{\tau}} p(z_x|z_{\mathcal{D}})\mathrm{d}z_x + \max_{\lambda,\lambda'} G(z_{\mathcal{D}},x,\lambda,\lambda') \\ &\leq \frac{1}{2}\log\left(2\pi\sigma_o^2\right)\int_{|z_x|\geq\widehat{\tau}} p(z_x|z_{\mathcal{D}})\mathrm{d}z_x \\ &+ \mathcal{O}\left(\frac{\sigma_o}{\sigma_n^2}\left(N^2\kappa^{2N}\tau+\sigma_o^2\right)\exp\left(-\frac{\tau^2}{2\sigma_o^2}\right)\right) \qquad \text{(C.18)}\end{aligned}$$

where $G(z_{\mathcal{D}},x,\lambda,\lambda')$ is previously defined in (C.11), the second inequality is due to

$$\begin{aligned}-\log p(z_x|z_{\mathcal{D}},\lambda) &= \frac{1}{2}\log\left(2\pi\sigma_{xx|\mathcal{D},\lambda}\right) + \frac{\left(z_x-\mu_{x|\mathcal{D},\lambda}\right)^2}{2\sigma_{xx|\mathcal{D},\lambda}} \\ &\leq \frac{1}{2}\log\left(2\pi\sigma_o^2\right) + \frac{\left(z_x-\mu_{x|\mathcal{D},\lambda}\right)^2}{2\sigma_{xx|\mathcal{D},\lambda}}\end{aligned}$$

with the inequality following from Lemma 7, and the last inequality in (C.18) holds

due to Lemma 15. Substituting (C.18) back into (C.17),

$$\begin{aligned}\mathbb{H}\left[Z_x|z_{\mathcal{D}}\right] - \widehat{\mathbb{H}}\left[\widehat{Z}_x|z_{\mathcal{D}}\right] &\leq \log\left(\frac{\sigma_o}{\sigma_n}\right)\int_{|z_x|\geq\widehat{\tau}} p(z_x|z_{\mathcal{D}})\,\mathrm{d}z_x \\ &+ \mathcal{O}\left(\frac{\sigma_o}{\sigma_n^2}\left(N^2\kappa^{2N}\tau+\sigma_o^2\right)\exp\left(-\frac{\tau^2}{2\sigma_o^2}\right)\right). \quad \text{(C.19)}\end{aligned}$$

By Lemma 7, since $\sigma_o \geq \sigma_n$, $\log(\sigma_o/\sigma_n) \geq 0$. Using Lemma 13,

$$\begin{aligned}\log\left(\frac{\sigma_o}{\sigma_n}\right)\int_{|z_x|\geq\widehat{\tau}} p(z_x|z_{\mathcal{D}})\,\mathrm{d}z_x &\leq 2\log\left(\frac{\sigma_o}{\sigma_n}\right)\Phi\left(-\frac{\tau}{\sigma_o}\right) \\ &\leq 2\log\left(\frac{\sigma_o}{\sigma_n}\right)\frac{\sigma_o}{\tau}\exp\left(-\frac{\tau^2}{2\sigma_o^2}\right) \\ &\leq 2\sigma_o\log\left(\frac{\sigma_o}{\sigma_n}\right)\exp\left(-\frac{\tau^2}{2\sigma_o^2}\right) \quad \text{(C.20)}\end{aligned}$$

where the second inequality follows from the Gaussian tail inequality (C.13), and the last inequality holds due to $\tau \geq 1$. Finally, by substituting (C.20) back into (C.19), Lemma 16 follows. □

C.11 Lemma 17

Lemma 17. *For all $z_{\mathcal{D}}$, $x \in \mathcal{X}\setminus\mathcal{D}$, $n = 1,\ldots,N$, $\gamma > 0$, and $\tau \geq 1$,*

$$|Q_n^*(z_{\mathcal{D}},x) - \widehat{Q}_n(z_{\mathcal{D}},x)| \leq \mathcal{O}\left(\sigma_o\tau\left(\frac{N^2\kappa^{2N}+\sigma_o^2}{\sigma_n^2} + N\log\frac{\sigma_o}{\sigma_n} + \log|\Lambda|\right)\exp\left(-\frac{\tau^2}{2\sigma_o^2}\right)\right)$$

where $\widehat{Q}_n(z_{\mathcal{D}},x)$ is previously defined in (B.4)*. Thus, by setting*

$$\tau = \mathcal{O}\left(\sigma_o\sqrt{\log\left(\frac{\sigma_o^2}{\gamma}\left(\frac{N^2\kappa^{2N}+\sigma_o^2}{\sigma_n^2} + N\log\frac{\sigma_o}{\sigma_n} + \log|\Lambda|\right)\right)}\right),$$

it directly follows that $|Q_n^*(z_{\mathcal{D}},x) - \widehat{Q}_n(z_{\mathcal{D}},x)| \leq \gamma/2$.

Proof. From (4.7) and (B.4),

$$\begin{aligned}\left|Q_n^*(z_{\mathcal{D}},x)-\widehat{Q}_n(z_{\mathcal{D}},x)\right| &\leq \left|\mathbb{H}\left[Z_x|z_{\mathcal{D}}\right]-\widehat{\mathbb{H}}\left[\widehat{Z}_x|z_{\mathcal{D}}\right]\right|+\int_{-\infty}^{-\widehat{\tau}}p(z_x|z_{\mathcal{D}})\Delta_{n+1}(z_x,-\widehat{\tau})\mathrm{d}z_x\\ &+ \int_{\widehat{\tau}}^{\infty}p(z_x|z_{\mathcal{D}})\Delta_{n+1}(z_x,\widehat{\tau})\mathrm{d}z_x\end{aligned}$$

where $\Delta_{n+1}(z_x,-\widehat{\tau})$ and $\Delta_{n+1}(z_x,\widehat{\tau})$ as

$$\begin{aligned}\Delta_{n+1}(z_x,-\widehat{\tau}) &\triangleq \left|V_{n+1}^*\left(z_{\mathcal{D}}\cup\{z_x\}\right)-V_{n+1}^*\left(z_{\mathcal{D}}\cup\{-\widehat{\tau}\}\right)\right|,\\ \Delta_{n+1}(z_x,\widehat{\tau}) &\triangleq \left|V_{n+1}^*\left(z_{\mathcal{D}}\cup\{z_x\}\right)-V_{n+1}^*\left(z_{\mathcal{D}}\cup\{\widehat{\tau}\}\right)\right|.\end{aligned}$$

Using Lemma 11, $\Delta_{n+1}(z_x,-\widehat{\tau})\leq(N-n)\log(\sigma_o/\sigma_n)+\log|\Lambda|\leq N\log(\sigma_o/\sigma_n)+\log|\Lambda|$. By a similar argument, $\Delta_{n+1}(z_x,\widehat{\tau})\leq N\log(\sigma_o/\sigma_n)+\log|\Lambda|$. Consequently,

$$\begin{aligned}\left|Q_n^*(z_{\mathcal{D}},x)-\widehat{Q}_n(z_{\mathcal{D}},x)\right| &\leq \left|\mathbb{H}[Z_x|z_{\mathcal{D}}]-\widehat{\mathbb{H}}[\widehat{Z}_x|z_{\mathcal{D}}]\right|\\ &+ \left(N\log\left(\frac{\sigma_o}{\sigma_n}\right)+\log|\Lambda|\right)\int_{|z_x|\geq\widehat{\tau}}p(z_x|z_{\mathcal{D}})\mathrm{d}z_x\\ &\leq \mathcal{O}\left(\sigma_o\tau\left(\frac{N^2\kappa^{2N}+\sigma_o^2}{\sigma_n^2}+N\log\frac{\sigma_o}{\sigma_n}+\log|\Lambda|\right)\exp\left(-\frac{\tau^2}{2\sigma_o^2}\right)\right).\end{aligned}$$

The last inequality follows from Lemmas 16 and 13 and the Gaussian tail inequality (C.13), which are applicable since $\tau\geq 1$.

To guarantee that $|Q_n^*(z_{\mathcal{D}},x)-\widehat{Q}_n(z_{\mathcal{D}},x)|\leq\gamma/2$, the value of τ to be determined must therefore satisfy the following inequality:

$$a\sigma_o\tau\left(\frac{N^2\kappa^{2N}+\sigma_o^2}{\sigma_n^2}+N\log\frac{\sigma_o}{\sigma_n}+\log|\Lambda|\right)\exp\left(-\frac{\tau^2}{2\sigma_o^2}\right) \leq \frac{\gamma}{2} \qquad \text{(C.21)}$$

where a is an existential constant[1] such that

$$|Q_n^*(z_{\mathcal{D}}, x) - \widehat{Q}_n(z_{\mathcal{D}}, x)| \;\leq\; a\sigma_o\tau\left(\frac{N^2\kappa^{2N}+\sigma_o^2}{\sigma_n^2} + N\log\frac{\sigma_o}{\sigma_n} + \log|\Lambda|\right)\exp\left(-\frac{\tau^2}{2\sigma_o^2}\right).$$

By taking log of both sides of (C.21),

$$\frac{\tau^2}{2\sigma_o^2} \;\geq\; \frac{1}{2}\log\left(\tau^2\right) + \log\left(\frac{2a\sigma_o}{\gamma}\left(\frac{N^2\kappa^{2N}+\sigma_o^2}{\sigma_n^2} + N\log\frac{\sigma_o}{\sigma_n} + \log|\Lambda|\right)\right). \quad \text{(C.22)}$$

Using the identity $\log(\tau^2) \leq \alpha\tau^2 - \log(\alpha) - 1$ with $\alpha = 1/(2\sigma_o^2)$, the RHS expression of (C.22) can be bounded from above by

$$\frac{\tau^2}{4\sigma_o^2} \;+\; \log\left(\frac{2\sqrt{2}a\sigma_o^2}{\sqrt{e}\gamma}\left(\frac{N^2\kappa^{2N} \;+\; \sigma_o^2}{\sigma_n^2} + N\log\frac{\sigma_o}{\sigma_n} \;+\; \log|\Lambda|\right)\right).$$

Hence, to satisfy (C.22), it suffices to determine the value of τ such that the following inequality holds:

$$\frac{\tau^2}{2\sigma_o^2} \;\geq\; \frac{\tau^2}{4\sigma_o^2} \;+\log\left(\frac{2\sqrt{2}a\sigma_o^2}{\sqrt{e}\gamma}\left(\frac{N^2\kappa^{2N}+\sigma_o^2}{\sigma_n^2} + N\log\frac{\sigma_o}{\sigma_n} + \log|\Lambda|\right)\right),$$

which implies

$$\begin{aligned}\tau \;&\geq\; 2\sigma_o\sqrt{\log\left(\frac{2\sqrt{2}a\sigma_o^2}{\sqrt{e}\gamma}\left(\frac{N^2\kappa^{2N}+\sigma_o^2}{\sigma_n^2} + N\log\frac{\sigma_o}{\sigma_n} + \log|\Lambda|\right)\right)} \\ &=\; \mathcal{O}\left(\sigma_o\sqrt{\log\left(\frac{\sigma_o^2}{\gamma}\left(\frac{N^2\kappa^{2N}+\sigma_o^2}{\sigma_n^2} + N\log\frac{\sigma_o}{\sigma_n} + \log|\Lambda|\right)\right)}\right)\end{aligned}$$

[1]Deriving an exact value for a should be straight-forward, albeit mathematically tedious, by taking into account the omitted constants in Lemmas 16 and 17.

Therefore, by setting

$$\tau \;=\; \mathcal{O}\left(\sigma_o\sqrt{\log\left(\frac{\sigma_o^2}{\gamma}\left(\frac{N^2\kappa^{2N}+\sigma_o^2}{\sigma_n^2}+N\log\frac{\sigma_o}{\sigma_n}+\log|\Lambda|\right)\right)}\right), \qquad \text{(C.23)}$$

$|Q_n^*(z_{\mathcal{D}},x)-\widehat{Q}_n(z_{\mathcal{D}},x)|\leq\gamma/2$ can be guaranteed. □

C.12 Lemma 18

Lemma 18. *For all $z_{\mathcal{D}}$ and $\lambda\in\Lambda$, let $\mathcal{A}$ and $\mathcal{B}$ denote subsets of sampling locations such that $\mathcal{A}\subseteq\mathcal{B}\subseteq\mathcal{X}$. Then, for all $x\in(\mathcal{X}\setminus\mathcal{B})\cup\mathcal{A}$,*

$$\mathbb{H}\left[Z_{\mathcal{A}\cup\{x\}}|z_{\mathcal{D}},\lambda\right]-\mathbb{H}\left[Z_{\mathcal{A}}|z_{\mathcal{D}},\lambda\right] \;\geq\; \mathbb{H}\left[Z_{\mathcal{B}\cup\{x\}}|z_{\mathcal{D}},\lambda\right]-\mathbb{H}\left[Z_{\mathcal{B}}|z_{\mathcal{D}},\lambda\right] .$$

Proof. If $x\in\mathcal{A}\subseteq\mathcal{B}$, $\mathbb{H}\left[Z_{\mathcal{A}\cup\{x\}}|z_{\mathcal{D}},\lambda\right]-\mathbb{H}\left[Z_{\mathcal{A}}|z_{\mathcal{D}},\lambda\right]=\mathbb{H}\left[Z_{\mathcal{B}\cup\{x\}}|z_{\mathcal{D}},\lambda\right]-\mathbb{H}\left[Z_{\mathcal{B}}|z_{\mathcal{D}},\lambda\right]=0$. Hence, this lemma holds trivially in this case. Otherwise, if $x\in\mathcal{X}\setminus\mathcal{B}$, we have

$$\begin{aligned}\mathbb{H}\left[Z_{\mathcal{A}\cup\{x\}}|z_{\mathcal{D}},\lambda\right]-\mathbb{H}\left[Z_{\mathcal{A}}|z_{\mathcal{D}},\lambda\right] &= \mathbb{E}\left[\mathbb{H}[Z_x|z_{\mathcal{D}}\cup Z_{\mathcal{A}},\lambda]|z_{\mathcal{D}},\lambda\right]\\ \mathbb{H}\left[Z_{\mathcal{B}\cup\{x\}}|z_{\mathcal{D}},\lambda\right]-\mathbb{H}\left[Z_{\mathcal{B}}|z_{\mathcal{D}},\lambda\right] &= \mathbb{E}\left[\mathbb{H}\left[Z_x|z_{\mathcal{D}}\cup Z_{\mathcal{B}},\lambda\right]|z_{\mathcal{D}},\lambda\right]\end{aligned}$$

from the chain rule for entropy. Let $\mathcal{A}'\triangleq\mathcal{B}\setminus\mathcal{A}\supseteq\emptyset$. Therefore, $\mathcal{B}$ can be re-written as $\mathcal{B}=\mathcal{A}\cup\mathcal{A}'$ where $\mathcal{A}\cap\mathcal{A}'=\emptyset$ (since $\mathcal{A}\subseteq\mathcal{B}$). Then,

$$\begin{aligned}\mathbb{H}\left[Z_{\mathcal{B}\cup\{x\}}|z_{\mathcal{D}},\lambda\right]-\mathbb{H}\left[Z_{\mathcal{B}}|z_{\mathcal{D}},\lambda\right] &= \mathbb{E}\left[\mathbb{H}\left[Z_x|z_{\mathcal{D}}\cup Z_{\mathcal{B}},\lambda\right]|z_{\mathcal{D}},\lambda\right]\\ &= \mathbb{E}\left[\mathbb{H}\left[Z_x|z_{\mathcal{D}}\cup Z_{\mathcal{A}}\cup Z_{\mathcal{A}'},\lambda\right]|z_{\mathcal{D}},\lambda\right]\\ &\leq \mathbb{E}\left[\mathbb{H}\left[Z_x|z_{\mathcal{D}}\cup Z_{\mathcal{A}},\lambda\right]|z_{\mathcal{D}},\lambda\right]\\ &= \mathbb{H}\left[Z_{\mathcal{A}\cup\{x\}}|z_{\mathcal{D}},\lambda\right]-\mathbb{H}\left[Z_{\mathcal{A}}|z_{\mathcal{D}},\lambda\right]\end{aligned}$$

where the inequality follows from the monotonicity of conditional entropy (i.e., "information never hurts" bound) [Cover and Thomas, 1991]. □

C.13 Lemma 19

Lemma 19. *For all $z_{\mathcal{D}}$ and $\lambda \in \Lambda$, let $\mathcal{S}^* \triangleq \arg\max_{\mathcal{S}\subseteq\mathcal{X}:|\mathcal{S}|=k} \mathbb{H}\left[Z_{\mathcal{S}}|z_{\mathcal{D}},\lambda\right]$. Then,*

$$\mathbb{H}\left[Z_{\mathcal{S}^*}|z_{\mathcal{D}},\lambda\right] \;\leq\; \frac{e}{e-1}\left(\mathbb{H}\left[Z_{\mathcal{S}_k^\lambda}|z_{\mathcal{D}},\lambda\right] + \frac{kr}{e}\right)$$

where $r = -\min(0, 0.5\log(2\pi e\sigma_n^2)) \geq 0$ and $\mathcal{S}_k^\lambda$ is the a priori greedy design previously defined in (4.18).

Proof. Let $\mathcal{S}^* \triangleq \{s_1^*,\ldots,s_k^*\}$ and $\mathcal{S}_i^\lambda \triangleq \{s_1^\lambda,\ldots,s_i^\lambda\}$ for $i = 1,\ldots,k$. Then,

$$\mathbb{H}\left[Z_{\mathcal{S}^*\cup\mathcal{S}_i^\lambda}|z_{\mathcal{D}},\lambda\right] \;=\; \mathbb{H}\left[Z_{\mathcal{S}^*}|z_{\mathcal{D}},\lambda\right] + \sum_{j=1}^{i}\left(\mathbb{H}\left[Z_{\mathcal{S}^*\cup\{s_1^\lambda,\ldots,s_j^\lambda\}}|z_{\mathcal{D}},\lambda\right] - \mathbb{H}\left[Z_{\mathcal{S}^*\cup\{s_1^\lambda,\ldots,s_{j-1}^\lambda\}}|z_{\mathcal{D}},\lambda\right]\right) \tag{C.24}$$

Clearly, if $s_j^\lambda \in \mathcal{S}^*$, $\mathbb{H}\left[Z_{\mathcal{S}^*\cup\{s_1^\lambda,\ldots,s_j^\lambda\}}|z_{\mathcal{D}},\lambda\right] - \mathbb{H}\left[Z_{\mathcal{S}^*\cup\{s_1^\lambda,\ldots,s_{j-1}^\lambda\}}|z_{\mathcal{D}},\lambda\right] = 0$. Otherwise, let $\widetilde{\mathcal{S}} \triangleq \mathcal{S}^*\cup\left\{s_1^\lambda,\ldots,s_{j-1}^\lambda\right\}$. Using the chain rule for entropy,

$$\begin{aligned}
\mathbb{H}\left[Z_{\widetilde{\mathcal{S}}\cup\{s_j^\lambda\}}|z_{\mathcal{D}},\lambda\right] - \mathbb{H}\left[Z_{\widetilde{\mathcal{S}}}|z_{\mathcal{D}},\lambda\right] &= \mathbb{E}\left[\mathbb{H}\left[Z_{s_j^\lambda}\middle| z_{\mathcal{D}}\cup Z_{\widetilde{\mathcal{S}}},\lambda\right]\middle| z_{\mathcal{D}},\lambda\right] \\
&\geq \mathbb{E}\left[\frac{1}{2}\log\left(2\pi e\sigma_n^2\right)\middle| z_{\mathcal{D}},\lambda\right] \\
&= \frac{1}{2}\log\left(2\pi e\sigma_n^2\right)
\end{aligned}$$

where the last inequality follows from Lemma 12. Combining these two cases and

using the fact that $r = -\min(0, 0.5\log(2\pi e\sigma_n^2))$,

$$\mathbb{H}\left[Z_{\widetilde{\mathcal{S}}\cup\{s_j^\lambda\}}|z_{\mathcal{D}},\lambda\right] - \mathbb{H}\left[Z_{\widetilde{\mathcal{S}}}|z_{\mathcal{D}},\lambda\right] \;\geq\; -r\ ,$$

which, by substituting back into (C.24), implies

$$\mathbb{H}\left[Z_{\mathcal{S}^*\cup\mathcal{S}_i^\lambda}|z_{\mathcal{D}},\lambda\right] \;\geq\; \mathbb{H}\left[Z_{\mathcal{S}^*}|z_{\mathcal{D}},\lambda\right] - ir\ . \tag{C.25}$$

Equivalently, (C.25) can be re-written as

$$\mathbb{H}\left[Z_{\mathcal{S}^*}|z_{\mathcal{D}},\lambda\right] \;\leq\; \mathbb{H}\left[Z_{\mathcal{S}^*\cup\mathcal{S}_i^\lambda}|z_{\mathcal{D}},\lambda\right] + ir\ . \tag{C.26}$$

On the other hand,

$$\begin{aligned}
\mathbb{H}\left[Z_{\mathcal{S}^*\cup\mathcal{S}_i^\lambda}|z_{\mathcal{D}},\lambda\right] &= \mathbb{H}\left[Z_{\mathcal{S}_i^\lambda}|z_{\mathcal{D}},\lambda\right] + \sum_{j=1}^{k}\left(\mathbb{H}\left[Z_{\mathcal{S}_i^\lambda\cup\{s_1^*,\ldots,s_j^*\}}|z_{\mathcal{D}},\lambda\right] - \mathbb{H}\left[Z_{\mathcal{S}_i^\lambda\cup\{s_1^*,\ldots,s_{j-1}^*\}}|z_{\mathcal{D}},\lambda\right]\right)\\
&\leq \mathbb{H}\left[Z_{\mathcal{S}_i^\lambda}|z_{\mathcal{D}},\lambda\right] + \sum_{j=1}^{k}\left(\mathbb{H}\left[Z_{\mathcal{S}_i^\lambda\cup\{s_j^*\}}|z_{\mathcal{D}},\lambda\right] - \mathbb{H}\left[Z_{\mathcal{S}_i^\lambda}|z_{\mathcal{D}},\lambda\right]\right)\\
&= \mathbb{H}\left[Z_{\mathcal{S}_i^\lambda}|z_{\mathcal{D}},\lambda\right] + \sum_{s\in\mathcal{S}^*}\left(\mathbb{H}\left[Z_{\mathcal{S}_i^\lambda\cup\{s\}}|z_{\mathcal{D}},\lambda\right] - \mathbb{H}\left[Z_{\mathcal{S}_i^\lambda}|z_{\mathcal{D}},\lambda\right]\right)\\
&\leq \mathbb{H}\left[Z_{\mathcal{S}_i^\lambda}|z_{\mathcal{D}},\lambda\right] + k\left(\mathbb{H}\left[Z_{\mathcal{S}_{i+1}^\lambda}|z_{\mathcal{D}},\lambda\right] - \mathbb{H}\left[Z_{\mathcal{S}_i^\lambda}|z_{\mathcal{D}},\lambda\right]\right)
\end{aligned}$$

where the first inequality is due to Lemma 18, and the last inequality follows from the construction of $\mathcal{S}_{i+1}^\lambda$(4.17). Combining this with (C.26),

$$\begin{aligned}
\mathbb{H}\left[Z_{\mathcal{S}^*}|z_{\mathcal{D}},\lambda\right] - \mathbb{H}\left[Z_{\mathcal{S}_i^\lambda}|z_{\mathcal{D}},\lambda\right] \;\leq\; & k\Big(\mathbb{H}\left[Z_{\mathcal{S}_{i+1}^\lambda}|z_{\mathcal{D}},\lambda\right]\\
& - \mathbb{H}\left[Z_{\mathcal{S}_i^\lambda}|z_{\mathcal{D}},\lambda\right]\Big) - i\min\left(0, \frac{1}{2}\log(2\pi e\sigma_n^2)\right).
\end{aligned}$$

Let $\delta_i \triangleq \mathbb{H}\left[Z_{\mathcal{S}^*}|z_{\mathcal{D}},\lambda\right] - \mathbb{H}\left[Z_{\mathcal{S}_i^\lambda}|z_{\mathcal{D}},\lambda\right]$. Then, the above inequality can be written concisely as

$$\delta_i \;\leq\; k(\delta_i - \delta_{i+1}) + ir \;,$$

which can consequently be cast as

$$\delta_{i+1} \;\leq\; \left(1 - \frac{1}{k}\right)\delta_i + \frac{ir}{k} \;. \tag{C.27}$$

Let $i \triangleq l-1$ and expand (C.27) recursively to obtain

$$\delta_l \;\leq\; \alpha^l \delta_0 + \frac{r}{k}\sum_{i=0}^{l-1}\alpha^i(l-i-1) \tag{C.28}$$

where $\alpha = 1 - 1/k$. To simplify the second term on the RHS expression of (C.28),

$$\begin{aligned}\sum_{i=0}^{l-1}\alpha^i(l-i-1) &= (l-1)\sum_{i=0}^{l-1}\alpha^i - \sum_{i=0}^{l-1} i\alpha^i \\ &= (l-1)\frac{1-\alpha^l}{1-\alpha} - \sum_{i=0}^{l-1} i\alpha^i \\ &= k(l-1)(1-\alpha^l) - \sum_{i=0}^{l-1} i\alpha^i \;. \end{aligned} \tag{C.29}$$

Then, let $\gamma_t \triangleq \sum_{i=0}^{t-1} i\alpha^i$ and $\phi_t \triangleq \sum_{i=1}^{t}\alpha^i$,

$$\begin{aligned}\gamma_{t+1} &= \sum_{i=0}^{t} i\alpha^i = \alpha\sum_{i=0}^{t-1}\alpha^i(i+1) = \alpha\left(\gamma_t + \sum_{i=0}^{t-1}\alpha^i\right) \\ &= \alpha\gamma_t + \sum_{i=0}^{t-1}\alpha^{i+1} = \alpha\gamma_t + \sum_{i=1}^{t}\alpha^i = \alpha\gamma_t + \phi_t \;. \end{aligned} \tag{C.30}$$

Expand (C.30) recursively to obtain

$$\begin{aligned}
\gamma_{t+1} &= \alpha^t \gamma_1 + \sum_{i=0}^{t-1} \alpha^i \phi_{t-i} \\
&= \sum_{i=0}^{t-1} \alpha^i \left(k(1-\alpha^{t-i+1}) - 1 \right) \\
&= \sum_{i=0}^{t-1} \alpha^i \left(k - k\alpha^{t-i+1} - 1 \right) \\
&= (k-1) \sum_{i=0}^{t-1} \alpha^i - k \sum_{i=0}^{t-1} \alpha^{t+1} \\
&= k(k-1)(1-\alpha^t) - kt\alpha^{t+1} \ . \qquad \text{(C.31)}
\end{aligned}$$

Let $t \triangleq l-1$. Substituting (C.31) back into (C.29),

$$\sum_{i=0}^{l-1} \alpha^i (l-i-1) = k(l-1) - k(k-1)(1-\alpha^{l-1}) \ .$$

Finally, let $l \triangleq k$. Substituting the above inequality back into (C.28),

$$\delta_k \leq \alpha^k \delta_0 + r(k-1)\alpha^{k-1} \ . \qquad \text{(C.32)}$$

Using the identity $1 - x \leq e^{-x}$,

$$\alpha^k = \left(1 - \frac{1}{k}\right)^k \leq \left(\exp\left(-\frac{1}{k}\right)\right)^k = \frac{1}{e} \ .$$

This directly implies

$$\alpha^{k-1} = \frac{\alpha^k}{\alpha} \leq \frac{1}{e\left(1-\frac{1}{k}\right)} \ .$$

Substituting these facts into (C.32),

$$\delta_k \;\leq\; \frac{\delta_0}{e} + \frac{kr}{e}\,,$$

which subsequently implies

$$\mathbb{H}\left[Z_{\mathcal{S}^*}|z_{\mathcal{D}},\lambda\right] - \mathbb{H}\left[Z_{\mathcal{S}_k^\lambda}|z_{\mathcal{D}},\lambda\right] \;\leq\; \frac{1}{e}\mathbb{H}\left[Z_{\mathcal{S}^*}|z_{\mathcal{D}},\lambda\right] + \frac{kr}{e}$$

or, equivalently,

$$\mathbb{H}\left[Z_{\mathcal{S}^*}|z_{\mathcal{D}},\lambda\right] \;\leq\; \frac{e}{e-1}\left(\mathbb{H}\left[Z_{\mathcal{S}_k^\lambda}|z_{\mathcal{D}},\lambda\right] + \frac{kr}{e}\right). \quad \square$$

Appendix D

Proofs of Main Results for Chapter 5

D.1 Proof of Theorem 10

Since $q(\mathbf{f}_n|\mathbf{f}_m)$ is set as the exact GP conditional, we can plug (5.10) into (5.9) to rewrite $\mathcal{L}_m(q)$ as

$$\begin{aligned}\mathcal{L}_m(q) &= \int p(\mathbf{f}_n|\mathbf{f}_m)\log\frac{p(\mathbf{f}_n,\mathbf{y}_n|\mathbf{f}_m)}{p(\mathbf{f}_n|\mathbf{f}_m)}\mathrm{d}\mathbf{f}_n \\ &= \int p(\mathbf{f}_n|\mathbf{f}_m)\log p(\mathbf{y}_n|\mathbf{f}_n)\mathrm{d}\mathbf{f}_n\ , \end{aligned} \tag{D.1}$$

where $p(\mathbf{y}_n|\mathbf{f}_n) \triangleq \mathcal{N}(\mathbf{y}_n|\mathbf{f}_n,\sigma_n^2\mathbf{I})$ (Section 5.1.2) and $p(\mathbf{f}_n|\mathbf{f}_m) \triangleq \mathcal{N}(\mathbf{f}_n|\mathbf{P}\mathbf{f}_m,\mathbf{K}_{nn}-\mathbf{Q}_{nn})$ as in (5.10). Plug these facts into (D.1), we obtain

$$\begin{aligned}\mathcal{L}_m(q) &= \mathbb{E}_{\mathbf{f}_n}\left[-\frac{1}{2\sigma_n^2}(\mathbf{y}_n-\mathbf{f}_n)^T(\mathbf{y}_n-\mathbf{f}_n)\right]+\text{const} \\ &= \mathbb{E}_{\mathbf{f}_n}\left[-\frac{1}{2\sigma_n^2}\mathbf{f}_n^T\mathbf{f}_n+\frac{1}{\sigma_n^2}\mathbf{f}_n^T\mathbf{y}_n\right]+\text{const}\ , \end{aligned} \tag{D.2}$$

where we absorbs terms that do not depend on $\mathbf{f}_n$ into const: Eq. (D.2) is derived by letting $(-1/2\sigma_n^2)\mathbf{y}_n^T\mathbf{y}_n$ be absorbed into const. Since $\mathbf{f}_n \sim \mathcal{N}(\mathbf{P}\mathbf{f}_m,\mathbf{K}_{nn}-\mathbf{Q}_{nn})$

according to (5.10), we can use the following Gaussian identities, $\mathbb{E}\left[\mathbf{f}_n\right] = \mathbf{P}\mathbf{f}_m$ and $\mathbb{E}\left[\mathbf{f}_n^T\mathbf{f}_n\right] = \mathbb{E}\left[\mathbf{f}_n\right]^T\mathbb{E}\left[\mathbf{f}_n\right] + \mathrm{tr}\left(\mathbf{K}_{nn} - \mathbf{Q}_{nn}\right)$, to rewrite $\mathcal{L}_m(q)$ as

$$\begin{aligned}\mathcal{L}_m(q) &= -\frac{1}{2\sigma_n^2}\mathbb{E}\left[\mathbf{f}_n^T\mathbf{f}_n\right] + \frac{1}{\sigma_n^2}\mathbb{E}\left[\mathbf{f}_n\right]^T\mathbf{y}_n + \text{const} \\ &= -\frac{1}{2\sigma_n^2}\mathbf{f}_m^T\mathbf{P}^T\mathbf{P}\mathbf{f}_m + \frac{1}{\sigma_n^2}\mathbf{f}_m^T\mathbf{P}^T\mathbf{y}_n + \text{const} \end{aligned} \tag{D.3}$$

where the last step is derived by applying the above Gaussian identities and letting $(-1/2\sigma_n^2)\mathrm{tr}\left(\mathbf{K}_{nn} - \mathbf{Q}_{nn}\right)$ be absorbed into const. Eq. (D.3) thus concludes our proof.

D.2 Proof of Theorem 11

Using Theorem 9, we can write $\mathcal{L}(q)$ as

$$\begin{aligned}\mathcal{L}(q) &= \int q(\mathbf{f}_m)\mathcal{L}_m(q)\mathrm{d}\mathbf{f}_m - \mathrm{KL}\left(q(\mathbf{f}_m)\|p(\mathbf{f}_m)\right) \\ &= \mathbb{E}_{\mathbf{f}_m}\left[-\frac{1}{2\sigma_n^2}\mathbf{f}_m^T\mathbf{P}^T\mathbf{P}\mathbf{f}_m + \frac{1}{\sigma_n^2}\mathbf{f}_m^T\mathbf{P}^T\mathbf{y}_n\right] \\ &\quad - \mathrm{KL}\left(q(\mathbf{f}_m)\|p(\mathbf{f}_m)\right) + \text{const}\ , \end{aligned} \tag{D.4}$$

where (D.4) is derived by plugging the RHS of (D.3) into $\mathcal{L}_m(q)$. Then, since $\mathbf{f}_m \sim \mathcal{N}\left(\boldsymbol{\mu}_+, \boldsymbol{\Sigma}_+\right)$, we can again plug the Gaussian identities $\mathbb{E}\left[\mathbf{f}_m\right] = \boldsymbol{\mu}_+$ and $\mathbb{E}\left[\mathbf{f}_m^T\mathbf{S}\mathbf{f}_m\right] = \boldsymbol{\mu}_+^T\mathbf{S}\boldsymbol{\mu}_+ + \mathrm{tr}\left(\mathbf{S}\boldsymbol{\Sigma}_+\right)$ with $\mathbf{S} = \mathbf{P}^T\mathbf{P}$ into (D.4) and further expand $\mathcal{L}(q)$ as a function of $\boldsymbol{\mu}_+$ and $\boldsymbol{\Sigma}_+$:

$$\begin{aligned}\mathcal{L}(q) &= -\frac{1}{2\sigma_n^2}\left(\boldsymbol{\mu}_+^T\mathbf{P}^T\mathbf{P}\boldsymbol{\mu}_+ + \mathrm{tr}\left(\mathbf{P}^T\mathbf{P}\boldsymbol{\Sigma}_+\right)\right) \\ &\quad + \frac{1}{\sigma_n^2}\boldsymbol{\mu}_+^T\mathbf{P}^T\mathbf{y}_n - \mathrm{KL}\left(q(\mathbf{f}_m)\|p(\mathbf{f}_m)\right) + \text{const}\ . \end{aligned} \tag{D.5}$$

On the other hand, recall that $p(\mathbf{f}_m) \triangleq \mathcal{N}\left(\mathbf{f}_m|\boldsymbol{\mu}_*, \boldsymbol{\Lambda}_*^{-1}\right)$ which implies

$$-\log p(\mathbf{f}_m) \;=\; \frac{1}{2}\left(\mathbf{f}_m - \boldsymbol{\mu}_*\right)^T \boldsymbol{\Lambda}_* \left(\mathbf{f}_m - \boldsymbol{\mu}_*\right) + \text{const} \;. \tag{D.6}$$

Then, we can represent $\mathrm{KL}\left(q(\mathbf{f}_m)\|p(\mathbf{f}_m)\right)$ as

$$\begin{aligned}
\mathrm{KL}\left(q(\mathbf{f}_m)\|p(\mathbf{f}_m)\right) &= \int q(\mathbf{f}_m)\log\frac{q(\mathbf{f}_m)}{p(\mathbf{f}_m)}\mathrm{d}\mathbf{f}_m \;=\; -\mathbb{H}\left[q(\mathbf{f}_m)\right] + \mathbb{E}_{\mathbf{f}_m}\left[-\log p(\mathbf{f}_m)\right] \\
&= \mathbb{E}\left[\frac{1}{2}\left(\mathbf{f}_m - \boldsymbol{\mu}_*\right)^T \boldsymbol{\Lambda}_* \left(\mathbf{f}_m - \boldsymbol{\mu}_*\right)\right] - \frac{1}{2}\log\left|\boldsymbol{\Sigma}_+\right| + \text{const} \\
&= \mathbb{E}\left[\frac{1}{2}\mathbf{f}_m^T\boldsymbol{\Lambda}_*\mathbf{f}_m - \boldsymbol{\mu}_*^T\boldsymbol{\Lambda}_*\mathbf{f}_m\right] - \frac{1}{2}\log\left|\boldsymbol{\Sigma}_+\right| + \text{const} \;.
\end{aligned} \tag{D.7}$$

In particular, the third equality follows directly from (D.6) and the fact that $\mathbb{H}\left[q(\mathbf{f}_m)\right] = (1/2)\log\left|\boldsymbol{\Sigma}_+\right| + \text{const}$ which is implied by $q(\mathbf{f}_m) \triangleq \mathcal{N}\left(\mathbf{f}_m|\boldsymbol{\mu}_+, \boldsymbol{\Sigma}_+\right)$. Then, we absorb $(1/2)\boldsymbol{\mu}_*^T\boldsymbol{\Lambda}_*\boldsymbol{\mu}_*$ into const since it does not depend on $\boldsymbol{\mu}_+$ and $\boldsymbol{\Sigma}_+$, thus arriving at (D.7). In addition, note that the expectation on the RHS of (D.7) is in fact over $q(\mathbf{f}_m)$ which is parameterized as a Gaussian. Therefore, we can once again invoke the Gaussian identities $\mathbb{E}\left[\mathbf{f}_m\right] = \boldsymbol{\mu}_+$ and $\mathbb{E}\left[\mathbf{f}_m^T\boldsymbol{\Lambda}_*\mathbf{f}_m\right] = \boldsymbol{\mu}_+^T\boldsymbol{\Lambda}_*\boldsymbol{\mu}_+ + \mathrm{tr}\left(\boldsymbol{\Lambda}_*\boldsymbol{\Sigma}_+\right)$ to simplify (D.7) as detailed below:

$$\begin{aligned}
\mathrm{KL}\left(q(\mathbf{f}_m)\|p(\mathbf{f}_m)\right) &= \frac{1}{2}\boldsymbol{\mu}_+^T\boldsymbol{\Lambda}_*\boldsymbol{\mu}_+ + \frac{1}{2}\mathrm{tr}\left(\boldsymbol{\Lambda}_*\boldsymbol{\Sigma}_+\right) \\
&- \frac{1}{2}\log\left|\boldsymbol{\Sigma}_+\right| - \boldsymbol{\mu}_+^T\boldsymbol{\Lambda}_*\boldsymbol{\mu}_* + \text{const} \;.
\end{aligned} \tag{D.8}$$

Thus, plugging (D.8) into (D.5) reveals (5.13) (Theorem 11), thus concluding our proof.

D.3 Proof of Theorem 13

As $\{i_l\}_{l=1}^r$ is sampled i.i.d from the uniform distribution over $\{1, 2, \ldots, p\}$, we have:

$$\begin{aligned}\mathbb{E}\left[\mathbf{F}(m, i_l)\right] &= \sum_{i=1}^{p} \Pr\left(i_l = i\right) \mathbf{F}(m, i) \;=\; \sum_{i=1}^{p} \frac{1}{p} \mathbf{F}(m, i) \\ &= \frac{1}{p} \sum_{i=1}^{p} \mathbf{F}(m, i) \; . \end{aligned} \tag{D.9}$$

Applying the above argument for $\mathbf{G}(m, i_l)$, we can obtain similar result for $\mathbf{G}$:

$$\mathbb{E}\left[\mathbf{G}(m, i_l)\right] = \frac{1}{p} \sum_{i=1}^{p} \mathbf{G}(m, i) \; . \tag{D.10}$$

Thus, (D.9) and (D.10) together imply

$$\begin{aligned}\mathbb{E}\left[\mathbf{G}(m, i_l) - \mathbf{F}(m, i_l)\boldsymbol{\mu}_+\right] &= \mathbb{E}\left[\mathbf{G}(m, i_l)\right] - \mathbb{E}\left[\mathbf{F}(m, i_l)\right]\boldsymbol{\mu}_+ \\ &= \frac{1}{p} \sum_{i=1}^{p} \left(\mathbf{G}(m, i) - \mathbf{F}(m, i)\boldsymbol{\mu}_+\right) \end{aligned} \tag{D.11}$$

Thus, taking the expectation over $\mathcal{S}$ for both sides of (5.22) and applying (D.11) to the resulting RHS, we obtain

$$\mathbb{E}_{\mathcal{S}}\left[\frac{\partial \widehat{\mathcal{L}}}{\partial \boldsymbol{\mu}_+}\right] = \mathbf{G}(m) + \sum_{i=1}^{p} \mathbf{G}(m, i) - \left(\mathbf{F}(m) + \sum_{i=1}^{p} \mathbf{F}(m, i)\right) \boldsymbol{\mu}_+ \; . \tag{D.12}$$

Now, plugging (5.20) and (5.21) into the RHS of (D.12), we have

$$\mathbb{E}_{\mathcal{S}}\left[\frac{\partial \widehat{\mathcal{L}}}{\partial \boldsymbol{\mu}_+}\right] = \boldsymbol{\Sigma}_{\mathbf{m}}^{-\mathbf{1}} \boldsymbol{\mu}_m - \boldsymbol{\Sigma}_m^{-1} \boldsymbol{\mu}_+ = \frac{\partial \mathcal{L}}{\partial \boldsymbol{\mu}_+} \; . \tag{D.13}$$

where the last equality follows from (5.17). Similarly, taking the expectation over $\mathcal{S}$ for both sides of (5.23) and applying (D.9) to the resulting RHS, we get

$$\begin{aligned}\mathbb{E}_{\mathcal{S}}\left[\frac{\partial\widehat{\mathcal{L}}}{\partial\boldsymbol{\Sigma}_{+}}\right] &= \frac{1}{2}\left(\boldsymbol{\Sigma}_{+}^{-1}-\left(\mathbf{F}(m)+\sum_{i=1}^{p}\mathbf{F}(m,i)\right)\right)\\ &= \frac{1}{2}\Big(\boldsymbol{\Sigma}_{+}^{-1}-\boldsymbol{\Sigma}_{m}^{-1}\Big)=\frac{\partial\mathcal{L}}{\partial\boldsymbol{\Sigma}_{+}}\ ,\end{aligned} \tag{D.14}$$

where the last two steps follow directly from (5.20) and (5.18), respectively. As such, (D.13) and (D.14) conclude our proof.

D.4 Proof of Theorem 14

The proof of Theorem 14 can be constructed by reiterating the exact arguments used in the proof of Theorem 13 (Appendix D.3). In particular, the results in (D.9) and (D.10) are reproduced here for convenience:

$$\mathbb{E}\left[\mathbf{F}(m,i_l)\right] = \frac{1}{p}\sum_{i=1}^{p}\mathbf{F}(m,i)\ , \tag{D.15}$$

$$\mathbb{E}\left[\mathbf{G}(m,i_l)\right] = \frac{1}{p}\sum_{i=1}^{p}\mathbf{G}(m,i)\ . \tag{D.16}$$

Thus, taking the expectation over $\mathcal{S}$ on both sides of (5.33) and applying (D.16) to the resulting RHS, we have

$$\mathbb{E}_{\mathcal{S}}\left[\frac{\partial\widehat{\mathcal{L}}}{\partial\boldsymbol{\eta}_1}\right] = -\left(\boldsymbol{\eta}_2-\boldsymbol{\eta}_1\boldsymbol{\eta}_1^{T}\right)^{-1}\boldsymbol{\eta}_1+\mathbf{G}(m)+\sum_{i=1}^{p}\mathbf{G}(m,i)\ . \tag{D.17}$$

Then, plugging (5.21) into the RHS of (D.17) recovers (5.31) which implies $\mathbb{E}_{\mathcal{S}}[\partial\widehat{\mathcal{L}}/\partial\boldsymbol{\eta}_1]=\partial\mathcal{L}/\partial\boldsymbol{\eta}_1$. Similarly, taking the expectation over $\mathcal{S}$ on both sides of (5.34) and applying

(D.16) to the resulting RHS, it follows that

$$\mathbb{E}_{\mathcal{S}}\left[\frac{\partial\widehat{\mathcal{L}}}{\partial\boldsymbol{\eta}_2}\right] \;=\; \frac{1}{2}\left(\boldsymbol{\eta}_2-\boldsymbol{\eta}_1\boldsymbol{\eta}_1^T\right)^{-1}-\frac{1}{2}\left(\mathbf{F}(m)+\sum_{i=1}^{p}\mathbf{F}(m,i)\right)\;. \qquad \text{(D.18)}$$

Finally, plugging (5.20) into the RHS of (D.18) recovers (5.32) which implies $\mathbb{E}_{\mathcal{S}}[\partial\widehat{\mathcal{L}}/\partial\boldsymbol{\eta}_2] = \partial\mathcal{L}/\partial\boldsymbol{\eta}_2$, thus completing our proof for Theorem 14.

Appendix E

Proofs of Auxiliary Results for Chapter 5

E.1 Proof of Lemma 6

$\forall\ \mathbf{f}_n, \mathbf{f}_m$ we have $p(\mathbf{y}_n) = p(\mathbf{f}_n, \mathbf{f}_m, \mathbf{y}_n)/p(\mathbf{f}_n, \mathbf{f}_m|\mathbf{y}_n)$ which directly implies

$$\log p(\mathbf{y}_n) \quad = \quad \log \frac{p(\mathbf{f}_n, \mathbf{f}_m, \mathbf{y}_n)}{p(\mathbf{f}_n, \mathbf{f}_m|\mathbf{y}_n)} \ . \tag{E.1}$$

Thus, let $q(\mathbf{f}_n, \mathbf{f}_m)$ be an arbitrary probability density function and integrate both side of (E.1) with q we have

$$\log p(\mathbf{y}_n) \quad = \quad \int q(\mathbf{f}_n, \mathbf{f}_m) \log \frac{p(\mathbf{f}_n, \mathbf{f}_m, \mathbf{y}_n)}{p(\mathbf{f}_n, \mathbf{f}_m|\mathbf{y}_n)} \mathrm{d}\mathbf{f}_n \mathrm{d}\mathbf{f}_m \ . \tag{E.2}$$

Then, using the identity $\log(ab) = \log a + \log b$, $\log\left(p(\mathbf{f}_n, \mathbf{f}_m, \mathbf{y}_n)/p(\mathbf{f}_n, \mathbf{f}_m|\mathbf{y}_n)\right)$ is decomposed as

$$\log \frac{p(\mathbf{f}_n, \mathbf{f}_m, \mathbf{y}_n)}{p(\mathbf{f}_n, \mathbf{f}_m|\mathbf{y}_n)} \quad = \quad \log \frac{p(\mathbf{f}_n, \mathbf{f}_m, \mathbf{y}_n)}{q(\mathbf{f}_n, \mathbf{f}_m)} + \log \frac{q(\mathbf{f}_n, \mathbf{f}_m)}{p(\mathbf{f}_n, \mathbf{f}_m|\mathbf{y}_n)}$$

Plug this into (E.2), we obtain

$$\begin{aligned}\log p(\mathbf{y}_n) &= \int q(\mathbf{f}_n, \mathbf{f}_m) \log \frac{p(\mathbf{f}_n, \mathbf{f}_m, \mathbf{y}_n)}{q(\mathbf{f}_n, \mathbf{f}_m)} \mathrm{d}\mathbf{f}_n \mathrm{d}\mathbf{f}_m \\ &+ \int q(\mathbf{f}_n, \mathbf{f}_m) \log \frac{q(\mathbf{f}_n, \mathbf{f}_m)}{p(\mathbf{f}_n, \mathbf{f}_m|\mathbf{y}_n)} \mathrm{d}\mathbf{f}_n \mathrm{d}\mathbf{f}_m \ . \end{aligned} \tag{E.3}$$

Using the definition of $\mathcal{L}(q)$ and KL(.‖.), (E.3) directly implies $\log p(\mathbf{y}_n) = \mathcal{L}(q) + \mathrm{KL}\left(q(\mathbf{f}_n, \mathbf{f}_m) \| p(\mathbf{f}_n, \mathbf{f}_m|\mathbf{y}_n)\right)$ which concludes our proof. □

E.2 Proof of Theorem 9

Let us rewrite $\mathcal{L}(q)$ (Lemma 6) as

$$\mathcal{L}(q) = \int_{\mathbf{f}_m} \int_{\mathbf{f}_n} q(\mathbf{f}_n, \mathbf{f}_m) \log \frac{p(\mathbf{f}_n, \mathbf{f}_m, \mathbf{y}_n)}{q(\mathbf{f}_n, \mathbf{f}_m)} \mathrm{d}\mathbf{f}_n \mathrm{d}\mathbf{f}_m \tag{E.4}$$

Then, using the facts that (a) $p(\mathbf{f}_n, \mathbf{f}_m, \mathbf{y}_n) = p(\mathbf{f}_n, \mathbf{y}_n|\mathbf{f}_m)p(\mathbf{f}_m)$ and (b) $q(\mathbf{f}_n, \mathbf{f}_m) = q(\mathbf{f}_n|\mathbf{f}_m)q(\mathbf{f}_m)$, we can decompose $\log\left(p(\mathbf{f}_n, \mathbf{f}_m, \mathbf{y}_n)/q(\mathbf{f}_n, \mathbf{f}_m)\right)$ as

$$\log \frac{p(\mathbf{f}_n, \mathbf{f}_m, \mathbf{y}_n)}{q(\mathbf{f}_n, \mathbf{f}_m)} = \log \frac{p(\mathbf{f}_n, \mathbf{y}_n|\mathbf{f}_m)}{q(\mathbf{f}_n|\mathbf{f}_m)} - \log \frac{q(\mathbf{f}_m)}{p(\mathbf{f}_m)} \tag{E.5}$$

Plug (E.5) into (E.4) and factorize $q(\mathbf{f}_n, \mathbf{f}_m)$ as in (b), we can rewrite $\mathcal{L}(q)$ as

$$\mathcal{L}(q) = \int_{\mathbf{f}_m} q(\mathbf{f}_m) \left[\int_{\mathbf{f}_n} q(\mathbf{f}_n|\mathbf{f}_m) \mathcal{V}(\mathbf{f}_n) \mathrm{d}\mathbf{f}_n \right] \mathrm{d}\mathbf{f}_m \ , \tag{E.6}$$

where we define $\mathcal{V}(\mathbf{f}_n) \triangleq \log\left(p(\mathbf{f}_n, \mathbf{y}_n|\mathbf{f}_m)/q(\mathbf{f}_n|\mathbf{f}_m)\right) - \log(q(\mathbf{f}_m)/p(\mathbf{f}_m))$. Thus, in order to simplify (E.6), we first evaluate

$$\begin{aligned}\mathcal{I}(\mathbf{f}_m) &\triangleq \int_{\mathbf{f}_n} q(\mathbf{f}_n|\mathbf{f}_m)\mathcal{V}(\mathbf{f}_n)\mathrm{d}\mathbf{f}_n \\ &= \int_{\mathbf{f}_n} q(\mathbf{f}_n|\mathbf{f}_m)\log\frac{p(\mathbf{f}_n, \mathbf{y}_n|\mathbf{f}_m)}{q(\mathbf{f}_n|\mathbf{f}_m)}\mathrm{d}\mathbf{f}_n - \log\frac{q(\mathbf{f}_m)}{p(\mathbf{f}_m)} \\ &= \mathcal{L}_m(q) - \log\frac{q(\mathbf{f}_m)}{p(\mathbf{f}_m)} \ . \end{aligned} \tag{E.7}$$

Plug (E.7) into (E.6), we obtain

$$\begin{aligned}\mathcal{L}(q) &= \int_{\mathbf{f}_m} q(\mathbf{f}_m)\left(\mathcal{L}_m(q) - \log\frac{q(\mathbf{f}_m)}{p(\mathbf{f}_m)}\right)\mathrm{d}\mathbf{f}_m \\ &= \int_{\mathbf{f}_m} q(\mathbf{f}_m)\mathcal{L}_m(q)\mathrm{d}\mathbf{f}_m - \mathrm{KL}\left(q(\mathbf{f}_m)\|p(\mathbf{f}_m)\right)\end{aligned} \tag{E.8}$$

which concludes our proof. □

E.3 Proof of Equation (5.4)

Marginalize out f_* from both sides of (5.2), we obtain

$$p(\mathbf{f}_n|\mathbf{f}_m) = \prod_{i=1}^{p} p(\mathbf{f}_i|\mathbf{f}_m) \ . \tag{E.9}$$

Then, using Bayes rule, $p(f_*|\mathbf{y}_n)$ can be expressed as

$$p(f_*|\mathbf{y}_n) = \int p(f_*|\mathbf{f}_m, \mathbf{y}_n)p(\mathbf{f}_m|\mathbf{y}_n)\mathrm{d}\mathbf{f}_m \ . \tag{E.10}$$

Hence, in order to prove (5.4), it suffices to show that $p(f_*|\mathbf{f}_m, \mathbf{y}_n) = p(f_*|\mathbf{f}_m, \mathbf{y}_p)$. To achieve this, note that $p(f_*|\mathbf{f}_m, \mathbf{y}_n)$ can be rewritten as

$$p(f_*|\mathbf{f}_m, \mathbf{y}_n) \;=\; \frac{p(f_*, \mathbf{y}_n|\mathbf{f}_m)}{p(\mathbf{y}_n|\mathbf{f}_m)} \;. \tag{E.11}$$

To simplify (E.11), we first factorize its denominator as

$$\begin{aligned} p(\mathbf{y}_n|\mathbf{f}_m) \;&=\; \int p(\mathbf{y}_n|\mathbf{f}_n)p(\mathbf{f}_n|\mathbf{f}_m)\mathrm{d}\mathbf{f}_n \;=\; \int \prod_{i=1}^{p} \Big(p(\mathbf{y}_i|\mathbf{f}_i)p(\mathbf{f}_i|\mathbf{f}_m)\mathrm{d}\mathbf{f}_i\Big) \\ &=\; \prod_{i=1}^{p} \left(\int p(\mathbf{y}_i|\mathbf{f}_i)p(\mathbf{f}_i|\mathbf{f}_m)\mathrm{d}\mathbf{f}_i\right) \;=\; \prod_{i=1}^{p} p(\mathbf{y}_i|\mathbf{f}_m) \;, \end{aligned} \tag{E.12}$$

where the second equality follows from (E.9) and the noise factorization $p(\mathbf{y}_n|\mathbf{f}_n) = \prod_{i=1}^{p} p(\mathbf{y}_i|\mathbf{f}_i)$. Likewise, the numerator on the RHS of (E.11) can be factorized as

$$\begin{aligned} p(f_*, \mathbf{y}_n|\mathbf{f}_m) \;&=\; \int p(f_*, \mathbf{f}_n|\mathbf{f}_m)p(\mathbf{y}_n|\mathbf{f}_n)\mathrm{d}\mathbf{f}_n \\ &=\; \left(\int p(f_*, \mathbf{f}_p|\mathbf{f}_m)p(\mathbf{y}_p|\mathbf{f}_p)\mathrm{d}\mathbf{f}_p\right) \times \left(\prod_{i=1}^{p-1} \int p(\mathbf{y}_i|\mathbf{f}_i)p(\mathbf{f}_i|\mathbf{f}_m)\mathrm{d}\mathbf{f}_i\right) \\ &=\; p(f_*, \mathbf{y}_p|\mathbf{f}_m) \left(\prod_{i=1}^{p-1} p(\mathbf{y}_i|\mathbf{f}_m)\right) \end{aligned} \tag{E.13}$$

where the second equality follows from the above factorization of $p(\mathbf{y}_n|\mathbf{f}_n)$ and (E.9). Thus, plugging (E.12) and (E.13) into (E.11) yields

$$p(f_*|\mathbf{f}_m, \mathbf{y}_n) \;=\; \frac{p(f_*, \mathbf{y}_p|\mathbf{f}_m)}{p(\mathbf{y}_p|\mathbf{f}_m)} \;=\; p(f_*|\mathbf{y}_p, \mathbf{f}_m) \;. \tag{E.14}$$

Plugging (E.14) into (E.10) concludes our proof of (5.4). □

E.4 Decomposable SGPs

This section demonstrates how the approximation $q^*(\mathbf{f}_m) \simeq p(\mathbf{f}_m|\mathbf{y}_n)$ employed in SoR, DTC, FITC and PITC [Quiñonero-Candela and Rasmussen, 2005] as well as FIC and PIC [Snelson and Ghahramani, 2007] can be decomposed to meet the conditions listed in (5.20) and (5.21). For clarity, the corresponding $q^*(\mathbf{f}_m)$ of these SGPs are first derived with respect to their approximated training and testing conditionals [Quiñonero-Candela and Rasmussen, 2005; Snelson and Ghahramani, 2007] in Appendix E.4.1.

E.4.1 Characterizing SGPs using (5.4)

To begin, we re-state the following exact expression of $p(f_*|\mathbf{y}_n)$, which is not subject to any assumption:

$$p(f_*|\mathbf{y}_n) = \int p(f_*|\mathbf{y}_n, \mathbf{f}_m)p(\mathbf{f}_m|\mathbf{y}_n)\mathrm{d}\mathbf{f}_m \ . \tag{E.15}$$

Then, we will demonstrate how this expression (E.15) can be simplified given the particular approximated training and testing conditionals of the existing SGPs.

E.4.1.1 PIC

Partially Independent Conditional (PIC) [Snelson and Ghahramani, 2007] jointly specifies its the approximated training and testing conditionals via the factorization in (5.2) which helps to simplify (E.15) as (Appendix E.3)

$$p(f_*|\mathbf{y}_n) = \int p(f_*|\mathbf{y}_p, \mathbf{f}_m)p(\mathbf{f}_m|\mathbf{y}_n)\mathrm{d}\mathbf{f}_m \ . \tag{E.16}$$

Since (5.2) is the only assumption made in PIC, it appears that its approximations $q^*(f_*|\mathbf{y}_p, \mathbf{f}_m)$ and $q^*(\mathbf{f}_m)$ in (5.4) coincide with $p(f_*|\mathbf{y}_p, \mathbf{f}_m)$ and $p(\mathbf{f}_m|\mathbf{y}_n)$. Thus, $q^*(\mathbf{f}_m)$ can be constructed by performing exact inference, assuming (5.2) holds, for $p(\mathbf{f}_m|\mathbf{y}_n)$. To do this, note that (5.2) also implies (E.9) which is re-stated here for convenience:

$$p(\mathbf{f}_n|\mathbf{f}_m) \;=\; \prod_{i=1}^{p} p(\mathbf{f}_i|\mathbf{f}_m) \;, \tag{E.17}$$

where $p(\mathbf{f}_i|\mathbf{f}_m)$ is the exact GP conditional [Rasmussen and Williams, 2006]:

$$p(\mathbf{f}_i|\mathbf{f}_m) \;\triangleq\; \mathcal{N}(\mathbf{f}_i|\mathbf{K}_{im}\mathbf{K}_{mm}^{-1}\mathbf{f}_m, \mathbf{K}_{ii} - \mathbf{Q}_{ii}) \tag{E.18}$$

$$\mathbf{Q}_{ii} \;\triangleq\; \mathbf{K}_{im}\mathbf{K}_{mm}^{-1}\mathbf{K}_{mi} \tag{E.19}$$

with $\mathbf{K}_{ii} \triangleq k(\mathbf{X}_i, \mathbf{X}_i)$, $\mathbf{K}_{im} \triangleq k(\mathbf{X}_i, \mathbf{U})$, $\mathbf{K}_{mi} \triangleq k(\mathbf{U}, \mathbf{X}_i)$ and $\mathbf{X}_i \triangleq \mathbf{X} \cap \mathbf{B}_i$. Using (E.17), $p(\mathbf{f}_n|\mathbf{f}_m)$ can be more compactly written as

$$p(\mathbf{f}_n|\mathbf{f}_m) \;=\; \mathcal{N}\Big(\mathbf{f}_n|\mathbf{K}_{nm}\mathbf{K}_{mm}\mathbf{f}_m, \mathbf{R}\Big) \;, \tag{E.20}$$

with $\mathbf{R} \triangleq \text{blkdiag}[\mathbf{K}_{11} - \mathbf{Q}_{11}, \ldots, \mathbf{K}_{pp} - \mathbf{Q}_{pp}] = \text{blkdiag}[\mathbf{K}_{nn} - \mathbf{Q}_{nn}]$ denotes the block diagonal matrix induced from the partition $\{\mathbf{X}_i\}_{i=1}^p$ of $\mathbf{X}$. Combining this with the fact that $p(\mathbf{f}_m) \triangleq \mathcal{N}(\mathbf{f}_m|\mathbf{0}_m, \mathbf{K}_{mm})$, the joint prior $p(\mathbf{f}_m, \mathbf{f}_n)$ can be expressed as

$$\mathcal{N}\left(\left(\begin{array}{c}\mathbf{f}_m\\ \mathbf{f}_n\end{array}\right)\middle|\left(\begin{array}{c}\mathbf{0}_m\\ \mathbf{0}_n\end{array}\right), \left(\begin{array}{cc}\mathbf{K}_{mm} & \mathbf{K}_{mn}\\ \mathbf{K}_{nm} & \mathbf{Q}_{nn} + \mathbf{R}\end{array}\right)\right) \tag{E.21}$$

Then, using the fact that $p(\mathbf{y}_n|\mathbf{f}_n) \triangleq \mathcal{N}(\mathbf{y}_n|\mathbf{f}_n, \sigma_n^2\mathbf{I})$, the joint prior $p(\mathbf{f}_m, \mathbf{y}_n)$ [Rasmussen and Williams, 2006] can be analytically derived as

$$\mathcal{N}\left(\begin{pmatrix}\mathbf{f}_m\\ \mathbf{y}_n\end{pmatrix}\middle|\begin{pmatrix}\mathbf{0}_m\\ \mathbf{0}_n\end{pmatrix}, \begin{pmatrix}\mathbf{K}_{mm} & \mathbf{K}_{mn}\\ \mathbf{K}_{nm} & \mathbf{Q}_{nn}+\mathbf{\Lambda}\end{pmatrix}\right) \tag{E.22}$$

with $\mathbf{\Lambda} \triangleq \text{blkdiag}[\mathbf{K}_{nn} - \mathbf{Q}_{nn} + \sigma_n^2\mathbf{I}]$. Thus, using (E.22), the conditional $p(\mathbf{f}_m|\mathbf{y}_n)$ is given as $\mathcal{N}(\mathbf{f}_m|\boldsymbol{\mu}_m, \mathbf{\Sigma}_m)$ where

$$\begin{aligned}\boldsymbol{\mu}_m &= \mathbf{K}_{mn}\left(\mathbf{Q}_{nn}+\mathbf{\Lambda}\right)^{-1}\mathbf{y}_n\ ,\\ \mathbf{\Sigma}_m &= \mathbf{K}_{mm} - \mathbf{K}_{mn}\left(\mathbf{Q}_{nn}+\mathbf{\Lambda}\right)^{-1}\mathbf{K}_{nm}\ .\end{aligned} \tag{E.23}$$

For computational efficiency, note that

$$\left(\mathbf{Q}_{nn}+\mathbf{\Lambda}\right)^{-1} = \mathbf{\Lambda}^{-1} - \mathbf{\Lambda}^{-1}\mathbf{K}_{nm}\mathbf{\Sigma}\mathbf{K}_{mn}\mathbf{\Lambda}^{-1}\ , \tag{E.24}$$

with $\mathbf{\Sigma} \triangleq \left(\mathbf{K}_{mm} + \mathbf{K}_{mn}\mathbf{\Lambda}^{-1}\mathbf{K}_{nm}\right)^{-1}$, which follows directly from the matrix inversion lemma. Then, multiply $\mathbf{K}_{mn}$ and $\mathbf{y}_n$ on both sides of (E.24), we have

$$\begin{aligned}\boldsymbol{\mu}_m &= \mathbf{K}_{mn}\left(\mathbf{Q}_{nn}+\mathbf{\Lambda}\right)^{-1}\mathbf{y}_n\\ &= \mathbf{K}_{mn}\left(\mathbf{\Lambda}^{-1} - \mathbf{\Lambda}^{-1}\mathbf{K}_{nm}\mathbf{\Sigma}\mathbf{K}_{mn}\mathbf{\Lambda}^{-1}\right)\mathbf{y}_n\\ &= \left(\mathbf{\Sigma}^{-1} - \mathbf{K}_{mn}\mathbf{\Lambda}^{-1}\mathbf{K}_{nm}\right)\mathbf{\Sigma}\mathbf{K}_{mn}\mathbf{\Lambda}^{-1}\mathbf{y}_n\\ &= \mathbf{K}_{mm}\mathbf{\Sigma}\mathbf{K}_{mn}\mathbf{\Lambda}^{-1}\mathbf{y}_n\ ,\end{aligned} \tag{E.25}$$

where the last equality follows from the definition of $\boldsymbol{\Sigma}$. Likewise, $\boldsymbol{\Sigma}_m$ is rewritten as

$$\begin{aligned}\boldsymbol{\Sigma}_m &= \mathbf{K}_{mm} - \mathbf{K}_{mn}\left(\mathbf{Q}_{nn} + \boldsymbol{\Lambda}\right)^{-1}\mathbf{K}_{nm} \\ &= \mathbf{K}_{mm} - \mathbf{K}_{mm}\boldsymbol{\Sigma}\mathbf{K}_{mn}\boldsymbol{\Lambda}^{-1}\mathbf{K}_{nm} \\ &= \mathbf{K}_{mm}\boldsymbol{\Sigma}\left(\boldsymbol{\Sigma}^{-1} - \mathbf{K}_{mn}\boldsymbol{\Lambda}^{-1}\mathbf{K}_{nm}\right) \\ &= \mathbf{K}_{mm}\boldsymbol{\Sigma}\mathbf{K}_{mm} \;, \end{aligned} \tag{E.26}$$

where the last equality again follows from the definition of $\boldsymbol{\Sigma}$. As such, the processing cost of evaluating $\boldsymbol{\mu}_m$ and $\boldsymbol{\Sigma}_m$ in (E.25) and (E.26) is only $\mathcal{O}(nm^2)$ instead of $\mathcal{O}(n^3)$ as incurred by (E.23). In particular, we can choose the partition $\{\mathbf{X}_i\}_{i=1}^p$ such that $p = \mathcal{O}(n/m)$ and the size of each partition $|\mathbf{X}_i| \leq \mathcal{O}(m)$ to guarantee the cost of evaluating

$$\mathbf{K}_{mn}\boldsymbol{\Lambda}^{-1}\mathbf{K}_{nm} = \sum_{i=1}^{p}\left(\mathbf{K}_{mi}\left(\mathbf{K}_{ii} - \mathbf{Q}_{ii} + \sigma_n^2\mathbf{I}\right)^{-1}\mathbf{K}_{im}\right)$$

is at most $\mathcal{O}\Big(\sum_{i=1}^{p}\left(m|\mathbf{X}_i|^2 + |\mathbf{X}_i|^3\right)\Big) = \mathcal{O}\left((n/m)m^3\right) = \mathcal{O}(nm^2)$. This directly implies $\boldsymbol{\Sigma}$ can be evaluated in $\mathcal{O}(nm^2 + m^3) = \mathcal{O}(nm^2)$ $(n > m)$. Finally, as $\boldsymbol{\Sigma}$ is m by m, it is easy to see that the cost of evaluating (E.25) and (E.26) is $\mathcal{O}(nm^2)$. The total processing cost of deriving $q^*(\mathbf{f}_m) \equiv p(\mathbf{f}_m|\mathbf{y}_n)$ is therefore $\mathcal{O}(nm^2)$.

E.4.1.2 PITC, FITC and FIC

The only difference between PITC [Quiñonero-Candela and Rasmussen, 2005] and PIC [Snelson and Ghahramani, 2007] is that the former assumes the following factorization in addition to that of (5.2)

$$p(f_*, \mathbf{f}_p|\mathbf{f}_m) = p(f_*|\mathbf{f}_m)p(\mathbf{f}_p|\mathbf{f}_m) \;, \tag{E.27}$$

which helps to further simplify $p(f_*|\mathbf{y}_p, \mathbf{f}_m) = p(f_*|\mathbf{f}_m)$ in (5.3). The training conditional $p(\mathbf{f}_n|\mathbf{f}_m)$ however remains the same as that of PIC (E.17) which implies PIC and PITC share the same approximation $q^*(\mathbf{f}_m) = p(\mathbf{f}_m|\mathbf{y}_n)$ as it is derived independently with the testing conditional $p(f_*|\mathbf{y}_p, \mathbf{f}_m)$. The processing cost of evaluating $q^*(\mathbf{f}_m)$ for PITC is thus the same as PIC's.

On the other hand, FITC appears to be a special case of PITC [Quiñonero-Candela and Rasmussen, 2005] when we replace blkdiag$[\mathbf{K}_{nn} - \mathbf{Q}_{nn}]$ by diag$[\mathbf{K}_{nn} - \mathbf{Q}_{nn}]$ in (E.20). Thus, it is easy to see that the corresponding $q^*(\mathbf{f}_m)$ of FITC can be derived by setting $\mathbf{\Lambda}^{-1} = \text{diag}[\mathbf{K}_{nn} - \mathbf{Q}_{nn} + \sigma_n^2\mathbf{I}]$ in the definition of $\mathbf{\Sigma}$ (E.24) and then, plugging it into (E.25) and (E.26). In terms of the processing cost, the only difference is that the complexity of evaluating $\mathbf{\Lambda}^{-1}$ for FITC is $\mathcal{O}(n)$ instead of $\mathcal{O}(nm^2)$. Nonetheless, computing $q^*(\mathbf{f}_m)$ still incurs $\mathcal{O}(nm^2)$ as the cost of evaluating $\mathbf{\Sigma}$ remains $\mathcal{O}(nm^2)$.

Finally, FIC [Snelson and Ghahramani, 2007] only differs from FITC when multiple tests $\mathbf{f}_* = [f_*^1, f_*^2, \ldots, f_*^k]^T$ are predicted simultaneously. Instead of retaining the exact testing conditional $p(\mathbf{f}_*|\mathbf{f}_m)$ like FITC, FIC assumes an additional factorization across the latent outputs of these testing inputs:

$$p(\mathbf{f}_*|\mathbf{f}_m) \;=\; \prod_{i=1}^{k} p(f_*^i|\mathbf{f}_m) \;. \tag{E.28}$$

However, this change only affects the testing conditional. The training conditional of FIC is thus the same as FITC's which implies they share the same $q^*(\mathbf{f}_m)$ which incurs the same $\mathcal{O}(nm^2)$ processing cost.

E.4.1.3 DTC and SoR

The only difference between Deterministic Training Conditional (DTC) [Seeger *et al.*, 2003] and PITC [Quiñonero-Candela and Rasmussen, 2005] is that the former replaces the exact block conditional $p(\mathbf{f}_i|\mathbf{f}_m) \triangleq \mathcal{N}(\mathbf{f}_i|\mathbf{K}_{im}\mathbf{K}_{mm}^{-1}\mathbf{f}_m, \mathbf{K}_{ii} - \mathbf{Q}_{ii})$ (E.18) with $\mathcal{N}(\mathbf{f}_i|\mathbf{K}_{im}\mathbf{K}_{mm}^{-1}\mathbf{f}_m, \mathbf{0})$. This effectively replaces $p(\mathbf{f}_n|\mathbf{f}_m) \triangleq \mathcal{N}(\mathbf{f}_n|\mathbf{K}_{nm}\mathbf{K}_{mm}^{-1}\mathbf{f}_m, \mathbf{K}_{nn} - \mathbf{Q}_{nn})$ by $\mathcal{N}(\mathbf{f}_n|\mathbf{K}_{nm}\mathbf{K}_{mm}^{-1}\mathbf{f}_m, \mathbf{0})$. Thus, the corresponding $q^*(\mathbf{f}_m)$ of DTC can be easily derived by changing that of PITC (hence, PIC) respectively. In particular, we can replace $\mathbf{K}_{nn} - \mathbf{Q}_{nn}$ with $\mathbf{0}$ in the definition of $\mathbf{\Lambda}$ (E.22), plug it into the expression of $\mathbf{\Sigma}$ (E.24) and consequently, (E.25) and (E.26) to finally derive DTC's $q^*(\mathbf{f}_m) = \mathcal{N}(\mathbf{f}_m|\boldsymbol{\mu}_m, \mathbf{\Sigma}_m)$ where

$$\begin{aligned} \boldsymbol{\mu}_m &= \frac{1}{\sigma_n^2}\mathbf{K}_{mm}\left(\mathbf{K}_{mm} + \frac{1}{\sigma_n^2}\mathbf{K}_{mn}\mathbf{K}_{nm}\right)^{-1}\mathbf{K}_{mn}\mathbf{y}_n \\ \mathbf{\Sigma}_m &= \mathbf{K}_{mm}\left(\mathbf{K}_{mm} + \frac{1}{\sigma_n^2}\mathbf{K}_{mn}\mathbf{K}_{nm}\right)^{-1}\mathbf{K}_{mm} \end{aligned} \tag{E.29}$$

Lastly, Subset of Regressors (SoR) [Smola and Bartlett, 2001] only differs from DTC [Seeger *et al.*, 2003] by replacing $p(f_*|\mathbf{f}_m) \triangleq \mathcal{N}(f_*|\mathbf{K}_{*m}\mathbf{K}_{mm}^{-1}\mathbf{f}_m, \mathbf{K}_{**} - \mathbf{Q}_{**})$ by $\mathcal{N}(f_*|\mathbf{K}_{*m}\mathbf{K}_{mm}^{-1}\mathbf{f}_m, 0)$ [Quiñonero-Candela and Rasmussen, 2005]. However, this change does not affect the training conditional of DTC which solely determines $q^*(\mathbf{f}_m)$ as argued previously in Appendix E.4.1.2. As a result, DTC and SoR share the same $q^*(\mathbf{f}_m)$. In terms of complexity, evaluating the inversion term in (E.29) incurs $\mathcal{O}(m^3 + nm^2) = \mathcal{O}(nm^2)$ which consequently implies the processing cost of (E.29) incurs $\mathcal{O}(nm^2)$.

E.4.2 Decomposability

This section demonstrate the decomposability of the existing SGP models as detailed in [Quiñonero-Candela and Rasmussen, 2005; Snelson and Ghahramani, 2007]. In

particular, our goal here is to show that their induced $q^*(\mathbf{f}_m)$ satisfy the Decomposability Conditions in (5.20) and (5.21). To simplify the analysis here, let us first recall that (a) PIC and PITC share the same $q^*(\mathbf{f}_m)$ (Appendix E.4.1.2), (b) SoR and DTC also induce the same $q^*(\mathbf{f}_m)$, and (c) FITC and FIC are special cases of PITC and PIC, respectively. Thus, it suffices to just demonstrate the decomposability of DTC (Appendix E.4.2.1) and PIC (Appendix E.4.2.2) here.

E.4.2.1 Decomposability of DTC

According to Appendix E.4.1.3, DTC's induced approximation $q^*(\mathbf{f}_m) = \mathcal{N}(\mathbf{f}_m|\boldsymbol{\mu}_m, \boldsymbol{\Sigma}_m)$ of $p(\mathbf{f}_m|\mathbf{y}_n)$ is given by

$$\boldsymbol{\mu}_m = \frac{1}{\sigma_n^2}\mathbf{K}_{mm}\boldsymbol{\Sigma}\mathbf{K}_{mn}\mathbf{y}_n \ , \tag{E.30}$$

$$\boldsymbol{\Sigma}_m = \mathbf{K}_{mm}\boldsymbol{\Sigma}\mathbf{K}_{mm} \ , \tag{E.31}$$

where $\boldsymbol{\Sigma} \triangleq ((1/\sigma_n^2)\mathbf{K}_{mn}\mathbf{K}_{nm} + \mathbf{K}_{mm})^{-1}$. Then, $\boldsymbol{\Sigma}_m^{-1}$ is decomposed as detailed below:

$$\begin{aligned}
\boldsymbol{\Sigma}_m^{-1} &= \mathbf{K}_{mm}^{-1}\boldsymbol{\Sigma}^{-1}\mathbf{K}_{mm}^{-1} \\
&= \mathbf{K}_{mm}^{-1}\left(\frac{1}{\sigma_n^2}\mathbf{K}_{mn}\mathbf{K}_{nm} + \mathbf{K}_{mm}\right)\mathbf{K}_{mm}^{-1} \\
&= \frac{1}{\sigma_n^2}\mathbf{K}_{mm}^{-1}\mathbf{K}_{mn}\mathbf{K}_{nm}\mathbf{K}_{mm}^{-1} + \mathbf{K}_{mm}^{-1} \ ,
\end{aligned} \tag{E.32}$$

where the first equality follows directly from (E.31). To decompose (E.32), suppose the data $\mathbf{D}$ is arbitrarily partitioned as $\mathbf{D} = \cup_{i=1}^p \mathbf{D}_i$ where $\mathbf{D}_i = (\mathbf{X}_i, \mathbf{y}_i)$ so that $\mathbf{X} = \cup_{i=1}^p \mathbf{X}_i$ and $\mathbf{y}_n = \left[\mathbf{y}_1^T, \mathbf{y}_2^T, \ldots, \mathbf{y}_p^T\right]^T$ (Section 5.2.2.1). Thus, $\mathbf{K}_{mn} = [\mathbf{k}_1, \mathbf{k}_2, \ldots, \mathbf{k}_p]$ where $\mathbf{k}_i$ denotes the sub-matrix which corresponds to $k(\mathbf{U}, \mathbf{X}_i)$ (Section 5.1.2). As

such, $\mathbf{K}_{mn}\mathbf{K}_{nm}$ can be decomposed as

$$\mathbf{K}_{mn}\mathbf{K}_{nm} \;=\; \sum_{i=1}^{p} \mathbf{k}_i \mathbf{k}_i^T \;. \tag{E.33}$$

Hence, plugging (E.33) into (E.32) we have

$$\boldsymbol{\Sigma}_m^{-1} \;=\; \sum_{i=1}^{p} \left(\frac{1}{\sigma_n^2} \mathbf{K}_{mm}^{-1} \mathbf{k}_i \mathbf{k}_i^T \mathbf{K}_{mm}^{-1} \right) + \mathbf{K}_{mm}^{-1} \;. \tag{E.34}$$

Also, plugging (E.30) and (E.31) into $\boldsymbol{\Sigma}_m^{-1}\boldsymbol{\mu}_m$ yields

$$\begin{aligned} \boldsymbol{\Sigma}_m^{-1}\boldsymbol{\mu}_m \;&=\; \frac{1}{\sigma_n^2} \mathbf{K}_{mm}^{-1} \mathbf{K}_{mn} \mathbf{y}_n \\ &=\; \sum_{i=1}^{p} \left(\frac{1}{\sigma_n^2} \mathbf{K}_{mm}^{-1} \mathbf{k}_i \mathbf{y}_i \right) \;, \end{aligned} \tag{E.35}$$

where the last equality is derived using the partitioned forms of $\mathbf{K}_{mn}$ and $\mathbf{y}_n$. Finally, setting $\mathbf{F}(m) \triangleq \mathbf{K}_{mm}^{-1}$, $\mathbf{F}(m,i) \triangleq (1/\sigma_n^2)\mathbf{K}_{mm}^{-1}\mathbf{k}_i\mathbf{k}_i^T\mathbf{K}_{mm}^{-1}$ in (E.34), $\mathbf{G}(m,i) \triangleq (1/\sigma_n^2)\mathbf{K}_{mm}^{-1}\mathbf{k}_i\mathbf{y}_i$ and $\mathbf{G}(m) = \mathbf{0}_{mm}$ in (E.35) reveals (5.20) and (5.21).

E.4.2.2 Decomposabiility of PIC

According to Appendix E.4.1.1, PIC [Snelson and Ghahramani, 2007] induces the following approximation $q^*(\mathbf{f}_m) = \mathcal{N}(\mathbf{f}_m|\boldsymbol{\mu}_m, \boldsymbol{\Sigma}_m)$ of $p(\mathbf{f}_m|\mathbf{y}_n)$:

$$\boldsymbol{\mu}_m \;=\; \mathbf{K}_{mm}\boldsymbol{\Sigma}\mathbf{K}_{mn}\boldsymbol{\Lambda}^{-1}\mathbf{y}_n \;, \tag{E.36}$$

$$\boldsymbol{\Sigma}_m \;=\; \mathbf{K}_{mm}\boldsymbol{\Sigma}\mathbf{K}_{mm} \;, \tag{E.37}$$

where $\boldsymbol{\Sigma} = (\mathbf{K}_{mm} + \mathbf{K}_{mn}\boldsymbol{\Lambda}^{-1}\mathbf{K}_{nm})^{-1}$, $\boldsymbol{\Lambda} = \text{blkdiag}\left[\boldsymbol{\Lambda}_1, \boldsymbol{\Lambda}_2, \ldots, \boldsymbol{\Lambda}_p\right]$ and the block matrix $\boldsymbol{\Lambda}_i = k(\mathbf{X}_i, \mathbf{X}_i) - \mathbf{K}_{im}\mathbf{K}_{mm}^{-1}\mathbf{K}_{mi} + \sigma_n^2\mathbf{I}$ with $\mathbf{K}_{im} \triangleq k(\mathbf{X}_i, \mathbf{U})$, $\mathbf{K}_{mi} \triangleq k(\mathbf{U}, \mathbf{X}_i)$, $\mathbf{X}_i = \mathbf{X} \cap \mathbf{B}_i$. Note that $\{\mathbf{B}_i\}_{i=1}^p$ denote PIC's partition of the input space as detailed

in (5.2). As such, we can equivalently write $\mathbf{\Lambda}^{-1} = \text{blkdiag}\left[\mathbf{\Lambda}_1^{-1}, \mathbf{\Lambda}_2^{-1}, \ldots, \mathbf{\Lambda}_p^{-1}\right]$ which implies

$$\begin{aligned} \mathbf{\Sigma}^{-1} &= \mathbf{K}_{mm} + \mathbf{K}_{mn}\mathbf{\Lambda}^{-1}\mathbf{K}_{nm} \\ &= \mathbf{K}_{mm} + \sum_{i=1}^{p} \mathbf{K}_{mi}\mathbf{\Lambda}_i^{-1}\mathbf{K}_{im} \ , \end{aligned} \tag{E.38}$$

where the last equality is derived by applying the partitioned forms of $\mathbf{\Lambda}^{-1}$ and $\mathbf{K}_{mn}$ to the RHS of the first equality. Then, plugging (E.38) into the first equality of (E.37) yields

$$\begin{aligned} \mathbf{\Sigma}_m^{-1} &= \mathbf{K}_{mm}^{-1}\mathbf{\Sigma}^{-1}\mathbf{K}_{mm}^{-1} \\ &= \mathbf{K}_{mm}^{-1} + \sum_{i=1}^{p} \left(\mathbf{K}_{mm}^{-1}\mathbf{K}_{mi}\mathbf{\Lambda}_i^{-1}\mathbf{K}_{im}\mathbf{K}_{mm}^{-1}\right) \end{aligned} \tag{E.39}$$

Thus, setting $\mathbf{F}(m) \triangleq \mathbf{K}_{mm}^{-1}$ and $\mathbf{F}(m,i) \triangleq \mathbf{K}_{mm}^{-1}\mathbf{K}_{mi}\mathbf{\Lambda}_i^{-1}\mathbf{K}_{im}\mathbf{K}_{mm}^{-1}$ in (E.39) reveals (5.20). On the other hand, we have

$$\begin{aligned} \mathbf{\Sigma}_m^{-1}\boldsymbol{\mu}_m &= \mathbf{K}_{mm}^{-1}\mathbf{K}_{mn}\mathbf{\Lambda}^{-1}\mathbf{y}_n \\ &= \mathbf{K}_{mm}^{-1}\left(\sum_{i=1}^{p} \mathbf{K}_{mi}\mathbf{\Lambda}^{-1}\mathbf{y}_i\right) \\ &= \sum_{i=1}^{p} \left(\mathbf{K}_{mm}^{-1}\mathbf{K}_{mi}\mathbf{\Lambda}^{-1}\mathbf{y}_i\right) \ , \end{aligned} \tag{E.40}$$

where the first equality follows directly from (E.36) and (E.37) while the second equality is derived using the partitioned forms of $\mathbf{K}_{mn}$, $\mathbf{\Lambda}^{-1}$ and $\mathbf{y}_n$ as mentioned above. Finally, setting $\mathbf{G}(m,i) \triangleq \mathbf{K}_{mm}^{-1}\mathbf{K}_{mi}\mathbf{\Lambda}^{-1}\mathbf{y}_i$ and $\mathbf{G}(m) = \mathbf{0}_{mm}$ in (E.40) reveals (5.21), thus completing our analysis.

E.4.3 Time Complexity of Evaluating F & G

Suppose that each partition of data has at most s data points, it follows directly from their definition in Appendices E.4.2.1 and E.4.2.2 that the processing cost of evaluating $\mathbf{F}(m,i)$ and $\mathbf{G}(m,i)$ is $\mathcal{O}(m^2 s)$. Thus, if the stochastic gradient is evaluated by sampling r partitions, the processing cost for each update iteration (Sections 5.2.2.1 and 5.2.2.2) is $\mathcal{O}(m^2 sr)$. In addition, there is an overhead $\mathcal{O}(m^3)$ time complexity incurred to evaluate $\mathbf{F}(m) \triangleq \mathbf{K}_{mm}^{-1}$ once and for all. This is $\mathcal{O}(m^3)$ in total if we specifically select the sampling size r and enforce the largest size $s_{\max}$ of each partition such that $rs_{\max} \leq m$. Hence, the total processing cost for k update iteration is $\mathcal{O}(km^3)$ which offers a significant computational advantage over the traditional $\mathcal{O}(nm^2)$ of SGPs [Quiñonero-Candela and Rasmussen, 2005] if we set $k = o(n/m)$. In practice, k can thus be used to control the trade-off between processing time and the approximation accuracy of $q^*(\mathbf{f}_m)$.

E.4.4 Time Complexity of Prediction

This section analyzes the cost of predicting $f^* \triangleq f(\mathbf{x}_*)$ using (5.4) and assuming that $\mathbf{x}_* \in \mathbf{B}_{p'}$ and $q(\mathbf{f}_m) = \mathcal{N}(\mathbf{f}_m|\boldsymbol{\mu}_+, \boldsymbol{\Sigma}_+)$ has already been evaluated using the techniques described in Sections 5.2.2.1 and 5.2.2.2. In particular, this includes the cost of (a) computing the approximated testing conditional $q(f_*|\mathbf{y}_{p'}, \mathbf{f}_m)$ and (b) analytically integrating it with $q(\mathbf{f}_m)$ (5.4) to evaluate $q(f_*) \simeq p(f_*|\mathbf{y}_n)$. To achieve this, we analytically derive $q(f_*)$ w.r.t $\boldsymbol{\mu}_+$ and $\boldsymbol{\Sigma}_+$ as detailed below.

E.4.4.1 PIC

Recall that the approximated testing conditional is set as the exact conditional $q(f_*|\mathbf{y}_{p'}, \mathbf{f}_m) = p(f_*|\mathbf{y}_{p'}, \mathbf{f}_m)$. To evaluate $p(f_*|\mathbf{y}_{p'}, \mathbf{f}_m)$, we use the fundamental definition of GP [Rasmussen and Williams, 2006] to state the following closed-form

expression for $p(f_*, \mathbf{f}_m, \mathbf{f}_{p'})$:

$$\mathcal{N}\left(\begin{bmatrix} f_* \\ \mathbf{f}_m \\ \mathbf{f}_{p'} \end{bmatrix} \middle| \mathbf{0}\,, \begin{bmatrix} \mathbf{K}_{**} & \mathbf{K}_{*m} & \mathbf{K}_{*p'} \\ \mathbf{K}_{m*} & \mathbf{K}_{mm} & \mathbf{K}_{mp'} \\ \mathbf{K}_{p'*} & \mathbf{K}_{p'm} & \mathbf{K}_{p'p'} \end{bmatrix}\right)\,, \tag{E.41}$$

where $\mathbf{K}_{p'p'} \triangleq k(\mathbf{X}_{p'}, \mathbf{X}_{p'})$, $\mathbf{K}_{p'*} \triangleq k(\mathbf{X}_{p'}, \mathbf{x}_*)$, $\mathbf{K}_{p'm} \triangleq k(\mathbf{X}_{p'}, \mathbf{U})$, $\mathbf{K}_{**} \triangleq k(\mathbf{x}_*, \mathbf{x}_*)$ and $\mathbf{K}_{*p'} \triangleq \mathbf{K}_{p'*}^T$, $\mathbf{K}_{mp'} \triangleq \mathbf{K}_{p'm}^T$, $\mathbf{X}_{p'} = \mathbf{X} \cap \mathbf{B}_{p'}$. Therefore, integrating (E.41) with $p(\mathbf{y}_{p'}|\mathbf{f}_{p'}) \triangleq \mathcal{N}(\mathbf{y}_{p'}|\mathbf{f}_{p'}, \sigma_n^2\mathbf{I})$, the closed-form expression for $p(f_*, \mathbf{f}_m, \mathbf{y}_{p'})$ is given below:

$$\mathcal{N}\left(\begin{bmatrix} f_* \\ \mathbf{f}_m \\ \mathbf{y}_{p'} \end{bmatrix} \middle| \mathbf{0}\,, \begin{bmatrix} \mathbf{K}_{**} & \mathbf{K}_{*m} & \mathbf{K}_{*p'} \\ \mathbf{K}_{m*} & \mathbf{K}_{mm} & \mathbf{K}_{mp'} \\ \mathbf{K}_{p'*} & \mathbf{K}_{p'm} & \mathbf{K}_{p'p'} + \sigma_n^2\mathbf{I} \end{bmatrix}\right)$$

Thus, the conditional $p(f_*|\mathbf{f}_m, \mathbf{y}_{p'})$ is analytically given as $\mathcal{N}(f_*|\mathbb{E}[f_*], \mathbb{V}[f_*])$ with $\mathbb{E}[f_*]$ and $\mathbb{V}[f_*]$ specified below using the conditional Gaussian identity:

$$\begin{aligned} \mathbb{E}[f_*] &= [\mathbf{K}_{*m}\ \mathbf{K}_{*p'}] \begin{bmatrix} \mathbf{K}_{mm} & \mathbf{K}_{mp'} \\ \mathbf{K}_{p'm} & \mathbf{K}_{p'p'} + \sigma_n^2\mathbf{I} \end{bmatrix}^{-1} \begin{bmatrix} \mathbf{f}_m \\ \mathbf{y}_{p'} \end{bmatrix} \\ \mathbb{V}[f_*] &= \mathbf{K}_{**} - [\mathbf{K}_{*m}\ \mathbf{K}_{*p'}] \begin{bmatrix} \mathbf{K}_{mm} & \mathbf{K}_{mp'} \\ \mathbf{K}_{p'm} & \mathbf{K}_{p'p'} + \sigma_n^2\mathbf{I} \end{bmatrix}^{-1} \begin{bmatrix} \mathbf{K}_{m*} \\ \mathbf{K}_{p'*} \end{bmatrix} \end{aligned}$$

To simplify the above expressions, let us denote

$$\mathbf{R} \triangleq \begin{bmatrix} \mathbf{K}_{mm} & \mathbf{K}_{mp'} \\ \mathbf{K}_{p'm} & \mathbf{K}_{p'p'} + \sigma_n^2\mathbf{I} \end{bmatrix}^{-1} = \begin{bmatrix} \mathbf{R}_{mm} & \mathbf{R}_{mp'} \\ \mathbf{R}_{p'm} & \mathbf{R}_{p'p'} \end{bmatrix}\,, \tag{E.42}$$

where $\mathbf{R}_{mm}$, $\mathbf{R}_{mp'}$, $\mathbf{R}_{p'm}$ and $\mathbf{R}_{p'p'}$ can be derived by applying the inversion lemma for partitioned matrices directly. Then, plugging (E.42) into the expression of $\mathbb{E}[f_*]$,

it can be simplified as

$$\mathbb{E}[f_*] \;=\; \mathbf{M}\mathbf{f}_m + \boldsymbol{\ell}\;, \tag{E.43}$$

with $\mathbf{M} \triangleq (\mathbf{K}_{*m}\mathbf{R}_{mm} + \mathbf{K}_{*p'}\mathbf{R}_{p'm})$ and $\boldsymbol{\ell} \triangleq (\mathbf{K}_{*m}\mathbf{R}_{mp'} + \mathbf{K}_{*p'}\mathbf{R}_{p'p'})\,\mathbf{y}_{p'}$. The testing conditional of PIC can thus be concisely written as $p(f_*|\mathbf{f}_m, \mathbf{y}_{p'}) = \mathcal{N}(f_*|\mathbf{M}\mathbf{f}_m + \boldsymbol{\ell}, \mathbb{V}[f_*]$. As such, $q(f_*)$ can be analytically derived by integrating $p(f_*|\mathbf{f}_m, \mathbf{y}_{p'})$ with $q(\mathbf{f}_m) = \mathcal{N}(\mathbf{f}_m|\boldsymbol{\mu}_+, \boldsymbol{\Sigma}_+)$:

$$q(f_*) \;=\; \int \mathcal{N}\left(f_*|\mathbf{M}\mathbf{f}_m + \boldsymbol{\ell}\right)\mathcal{N}(\mathbf{f}_m|\boldsymbol{\mu}_+, \boldsymbol{\Sigma}_+)\mathrm{d}\mathbf{f}_m \;=\; \mathcal{N}\left(f_*|\boldsymbol{\mu}_*, \boldsymbol{\Sigma}_*\right)\;, \tag{E.44}$$

with $\boldsymbol{\mu}_* = \mathbf{M}\boldsymbol{\mu}_+ + \boldsymbol{\ell}$ and $\boldsymbol{\Sigma}_* = \mathbb{V}[f_*] + \mathbf{M}\boldsymbol{\Sigma}_+\mathbf{M}^T$. Eq. (E.44) thus represents PIC's predictive distribution at $\mathbf{x}_*$. Its prediction is given by the distribution's mean $\boldsymbol{\mu}_* = \mathbf{M}\boldsymbol{\mu}_+ + \boldsymbol{\ell}$ which can otherwise be written as

$$\boldsymbol{\mu}_* \;=\; [\mathbf{K}_{*m}\;\mathbf{K}_{*p'}] \begin{bmatrix} \mathbf{R}_{mm} & \mathbf{R}_{mp'} \\ \mathbf{R}_{p'm} & \mathbf{R}_{p'p'} \end{bmatrix} \begin{bmatrix} \boldsymbol{\mu}_+ \\ \mathbf{y}_{p'} \end{bmatrix}\;, \tag{E.45}$$

by plugging in the above definitions of $\mathbf{M}$ and $\boldsymbol{\ell}$. Then, assuming $\mathbf{R}\left[\boldsymbol{\mu}_+^T \;\; \mathbf{y}_{p'}^T\right]^T$ are precomputed in advance for every block $p' = 1, 2, \ldots, p$, the single-input prediction cost (e.g., the cost of evaluating (E.45)) for any single input $\mathbf{x}_* \in \mathbf{B}_{p'}$ is at most $\mathcal{O}(m + |\mathbf{X}_{p'}|)$. In particular, if the input space is partitioned so that $|\mathbf{X}_{p'}| \leq \mathcal{O}(m)$ for any $p' = 1, 2, \ldots, p$, precomputing $\mathbf{R}\left[\boldsymbol{\mu}_+^T \;\; \mathbf{y}_{p'}^T\right]^T$ for all blocks incurs $\mathcal{O}(pm^3)$ and $\mathcal{O}(pm)$ for time and storage complexity, respectively. The overall predicting cost is then reduced to $\mathcal{O}(m)$ accordingly.

E.4.4.2 PITC, FITC, FIC and DTC

Since we only analyze the complexity of single-input prediction at $\mathbf{x}_*$, the approximated testing conditionals of PITC, FITC, FIC and DTC are set to the same exact conditional of GP [Quiñonero-Candela and Rasmussen, 2005]:

$$q(f_*|\mathbf{f}_m, \mathbf{y}_{p'}) \;=\; \mathcal{N}(f_* \mid \mathbf{P}\mathbf{f}_m,\; \mathbf{K}_{**} - \mathbf{Q}_{**}) \;, \tag{E.46}$$

where $\mathbf{P} \triangleq \mathbf{K}_{*m}\mathbf{K}_{mm}^{-1}$ and $\mathbf{Q}_{**} \triangleq \mathbf{K}_{*m}\mathbf{K}_{mm}^{-1}\mathbf{K}_{m*}$. As such, their predictive distributions are generally given by

$$\begin{aligned} q(f_*) \;&=\; \int \mathcal{N}(f_*|\mathbf{P}\mathbf{f}_m, \mathbf{K}_{**} - \mathbf{Q}_{**})\mathcal{N}(\mathbf{f}_m|\boldsymbol{\mu}_+, \boldsymbol{\Sigma}_+)\mathrm{d}\mathbf{f}_m \\ &=\; \mathcal{N}(f_* \mid \mathbf{P}\boldsymbol{\mu}_+,\; \mathbf{K}_{**} - \mathbf{Q}_{**} + \mathbf{P}\boldsymbol{\Sigma}_+\mathbf{P}^T) \;. \end{aligned} \tag{E.47}$$

From (E.47) and the above definition of $\mathbf{P}$, the prediction at $\mathbf{x}_*$ is explicitly given as

$$\boldsymbol{\mu}_* \;=\; \mathbf{K}_{*m}\mathbf{K}_{mm}^{-1}\boldsymbol{\mu}_+ \;, \tag{E.48}$$

which can be evaluated in $\mathcal{O}(m)$ assuming that $\mathbf{K}_{mm}^{-1}\boldsymbol{\mu}_+$ is precomputed in advance incurring $\mathcal{O}(m^3)$ and $\mathcal{O}(m)$ for time and storage complexity, respectively.

E.4.4.3 SoR

SoR [Smola and Bartlett, 2001] further simplifies the testing conditional in (E.46) by additionally imposing a deterministic relationship between f_* and $\mathbf{f}_m$:

$$q(f_*|\mathbf{f}_m, \mathbf{y}_{p'}) \;=\; \mathcal{N}(f_* \mid \mathbf{P}\mathbf{f}_m, \mathbf{0}) \;. \tag{E.49}$$

Integrating the RHS of (E.49) with $q(\mathbf{f}_m) = \mathcal{N}(\mathbf{f}_m|\boldsymbol{\mu}_+, \boldsymbol{\Sigma}_+)$ results in SoR predictive distribution which is given by

$$\begin{aligned} q(f_*) &= \int \mathcal{N}(f_*|\mathbf{P}\mathbf{f}_m, \mathbf{0})\mathcal{N}(\mathbf{f}_m|\boldsymbol{\mu}_+, \boldsymbol{\Sigma}_+)\mathrm{d}\mathbf{f}_m \\ &= \mathcal{N}(f_* \mid \mathbf{P}\boldsymbol{\mu}_+,\ \mathbf{P}\boldsymbol{\Sigma}_+\mathbf{P}^T)\ . \end{aligned} \tag{E.50}$$

Thus, similar to PITC, FITC, FIC and DTC, SoR's prediction also shares the same form of $\mathbf{K}_{*m}\mathbf{K}_{mm}^{-1}\boldsymbol{\mu}_+$ which can be evaluated in $\mathcal{O}(m)$ assuming that $\mathbf{K}_{mm}^{-1}\boldsymbol{\mu}_+$ is precomputed prior to prediction.

E.5 The Canonical Parameterization of Gaussian Distributions

This section features the canonical parameterization of Gaussian distribution and highlights some of its properties which have been previously used in Section 5.2.2.2. For ease of reading, we demonstrate this in the context of $q(\mathbf{f}_m)$ which is originally specified using the moment parameterization $q(\mathbf{f}_m) \triangleq \mathcal{N}(\mathbf{f}_m|\boldsymbol{\mu}_+, \boldsymbol{\Sigma}_+)$. In particular, let $\boldsymbol{\theta} \triangleq [\boldsymbol{\theta}_1; \mathrm{vec}(\boldsymbol{\theta}_2)]$ where $\boldsymbol{\theta}_1 \triangleq \boldsymbol{\Sigma}_+^{-1}\boldsymbol{\mu}_+$ and $\boldsymbol{\theta}_2 \triangleq -(1/2)\boldsymbol{\Sigma}_+^{-1}$, we begin with the following re-parameterization of $q(\mathbf{f}_m|\boldsymbol{\theta})$ with respect to $\boldsymbol{\theta}$:

$$q(\mathbf{f}_m|\boldsymbol{\theta}) = \mathcal{N}(\mathbf{f}_m|\boldsymbol{\mu}_+, \boldsymbol{\Sigma}_+) = \mathbf{h}(\mathbf{f}_m)\exp\left(\boldsymbol{\theta}^T\mathbf{T}(\mathbf{f}_m) - \mathbf{A}(\boldsymbol{\theta})\right) \tag{E.51}$$

where $\mathbf{T}(\mathbf{f}_m) \triangleq \left[\mathbf{f}_m; \mathrm{vec}(\mathbf{f}_m\mathbf{f}_m^T)\right]$, $\mathbf{h}(\mathbf{f}_m) \triangleq (2\pi)^{-m/2}$ and the normalizing function $\mathbf{A}(\boldsymbol{\theta})$ is defined as

$$\begin{aligned} \mathbf{A}(\boldsymbol{\theta}) &\triangleq -\frac{1}{2}\mathrm{tr}\left(\boldsymbol{\theta}_2\boldsymbol{\theta}_1\boldsymbol{\theta}_1^T\right) - \frac{1}{2}\log|-2\boldsymbol{\theta}_2| \\ &= \frac{1}{2}\boldsymbol{\mu}_+^T\boldsymbol{\Sigma}_+^{-1}\boldsymbol{\mu}_+ + \frac{1}{2}\log\left|\boldsymbol{\Sigma}_+^{-1}\right|\ . \end{aligned} \tag{E.52}$$

Alternatively, we can define $\mathbf{Z}(\theta)$ such that $\mathbf{A}\left(\boldsymbol{\theta}\right) = \log \mathbf{Z}\left(\boldsymbol{\theta}\right)$ and rewrite (E.51) as

$$q(\mathbf{f}_m) \;=\; \frac{1}{\mathbf{Z}(\boldsymbol{\theta})}\mathbf{h}(\mathbf{f}_m)\,\exp\left(\boldsymbol{\theta}^T\mathbf{T}(\mathbf{f}_m)\right)\;. \tag{E.53}$$

Since $\int q(\mathbf{f}_m)\mathrm{d}\mathbf{f}_m = 1$, (E.53) effectively implies

$$\mathbf{Z}\left(\boldsymbol{\theta}\right) \;=\; \int \mathbf{h}(\mathbf{f}_m)\exp\left(\boldsymbol{\theta}^T\mathbf{T}(\mathbf{f}_m)\right)\mathrm{d}\mathbf{f}_m\;. \tag{E.54}$$

Then, using these establishments, we are now able to verify the identities employed in Section 5.2.2.1 as detailed in the below subsections.

E.5.1 Evaluating $\boldsymbol{\eta} \triangleq \mathbb{E}\left[\mathbf{T}(\mathbf{f}_m)\right]$

This section will demonstrate how $\mathbb{E}\left[\mathbf{T}(\mathbf{f}_m)\right]$ can be derived as a function of $\boldsymbol{\mu}_+$ and $\boldsymbol{\Sigma}_+$. To achieve this, we first use the definition of $\mathbf{T}(\mathbf{f}_m)$ to obtain

$$\begin{aligned}\mathbb{E}\left[\mathbf{T}(\mathbf{f}_m)\right] &= \left[\mathbb{E}\left[\mathbf{f}_m\right];\mathbb{E}\left[\mathrm{vec}\left(\mathbf{f}_m\mathbf{f}_m^T\right)\right]\right]\\ &= \left[\mathbb{E}\left[\mathbf{f}_m\right];\mathrm{vec}\left(\mathbb{E}\left[\mathbf{f}_m\mathbf{f}_m^T\right]\right)\right]\\ &= \left[\boldsymbol{\mu}_+;\mathrm{vec}\left(\boldsymbol{\mu}_+\boldsymbol{\mu}_+^T+\boldsymbol{\Sigma}_+\right)\right]\;,\end{aligned} \tag{E.55}$$

where the second step follows from the definition of vec and expectation of vector while the last step is derived using the Gaussian identities $\mathbb{E}[\mathbf{f}_m] = \boldsymbol{\mu}_+$ and $\mathbb{E}[\mathbf{f}_m\mathbf{f}_m^T] = \boldsymbol{\mu}_+\boldsymbol{\mu}_+^T + \boldsymbol{\Sigma}_+$ (since $\mathbf{f}_m \sim \mathcal{N}(\boldsymbol{\mu}_+, \boldsymbol{\Sigma}_+)$). In particular, if we define $\boldsymbol{\eta}_1 \triangleq \boldsymbol{\mu}_+$ and $\boldsymbol{\eta}_2 \triangleq \boldsymbol{\mu}_+\boldsymbol{\mu}_+^T + \boldsymbol{\Sigma}_+$, then $\boldsymbol{\eta} \triangleq [\boldsymbol{\eta}_1; \mathrm{vec}(\boldsymbol{\eta}_2)] = \mathbb{E}\left[\mathbf{T}(\mathbf{f}_m)\right]$ denotes the expectation parameters $q(\mathbf{f}_m)$ in its canonical parameterization.

E.5.2 Evaluating $\mathbf{G}(\boldsymbol{\theta})$

This section focuses on explicitly representing the Fisher information $\mathbf{G}(\boldsymbol{\theta})$ in terms of $\mathbf{A}(\boldsymbol{\theta})$. To achieve this, we note that

$$\begin{aligned} \frac{\partial \log q(\mathbf{f}_m|\boldsymbol{\theta})}{\partial \boldsymbol{\theta}} &= \frac{\partial}{\partial \boldsymbol{\theta}}\Big(\boldsymbol{\theta}^T\mathbf{T}(\mathbf{f}_m) - \mathbf{A}(\boldsymbol{\theta})\Big) \\ &= \mathbf{T}(\mathbf{f}_m) - \frac{\partial \mathbf{A}(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}} \ . \end{aligned} \tag{E.56}$$

Thus, applying the identity $\partial f(x)/\partial x^T = (\partial f(x)/\partial x)^T$ to (E.56), we have

$$\begin{aligned} \frac{\partial \log q(\mathbf{f}_m|\boldsymbol{\theta})}{\partial \boldsymbol{\theta}^T} &= \mathbf{T}(\mathbf{f}_m)^T - \left(\frac{\partial \mathbf{A}(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}}\right)^T \\ &= \mathbf{T}(\mathbf{f}_m)^T - \frac{\partial \mathbf{A}(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}^T} \ . \end{aligned} \tag{E.57}$$

Then, differentiate both sides of (E.57) with respect to $\boldsymbol{\theta}$ yields

$$\frac{\partial^2 \log q(\mathbf{f}_m|\boldsymbol{\theta})}{\partial \boldsymbol{\theta}\partial \boldsymbol{\theta}^T} = -\frac{\partial^2 \mathbf{A}(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}\partial \boldsymbol{\theta}^T} \ . \tag{E.58}$$

Plugging (E.58) into the definition of $\mathbf{G}(\boldsymbol{\theta})$ reveals (5.25).

E.5.3 Proof of $\partial\boldsymbol{\eta}/\partial\boldsymbol{\theta} = \mathbf{G}(\boldsymbol{\theta})$

Let us differentiate $\mathbf{A}(\boldsymbol{\theta})$ with respect to $\boldsymbol{\theta}$:

$$\begin{aligned} \frac{\partial \mathbf{A}(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}} &= \frac{\partial \log \mathbf{Z}(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}} = \frac{1}{\mathbf{Z}(\boldsymbol{\theta})}\frac{\partial \mathbf{Z}(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}} \\ &= \frac{1}{\mathbf{Z}(\boldsymbol{\theta})}\int \mathbf{h}(\mathbf{f}_m)\frac{\partial}{\partial \boldsymbol{\theta}}\exp\left(\boldsymbol{\theta}^T\mathbf{T}(\mathbf{f}_m)\right)\mathrm{d}\mathbf{f}_m \\ &= \frac{1}{\mathbf{Z}(\boldsymbol{\theta})}\int \mathbf{h}(\mathbf{f}_m)\exp\left(\boldsymbol{\theta}^T\mathbf{T}(\mathbf{f}_m)\right)\mathbf{T}(\mathbf{f}_m)\mathrm{d}\mathbf{f}_m \\ &= \int q(\mathbf{f}_m|\boldsymbol{\theta})\mathbf{T}(\mathbf{f}_m)\mathrm{d}\mathbf{f}_m = \mathbb{E}\left[\mathbf{T}(\mathbf{f}_m)\right] = \boldsymbol{\eta} \ . \end{aligned} \tag{E.59}$$

Thus, (E.59) effectively implies $\boldsymbol{\eta}^T = \partial \mathbf{A}(\boldsymbol{\theta})/\partial \boldsymbol{\theta}^T$. Hence, differentiating both side of this equality with respect to $\boldsymbol{\theta}$ yields $\partial \boldsymbol{\eta}^T/\partial \boldsymbol{\theta} = \partial^2 \mathbf{A}(\boldsymbol{\theta})/\partial \boldsymbol{\theta} \partial \boldsymbol{\theta}^T = \mathbf{G}(\boldsymbol{\theta})$. Since $\mathbf{G}(\boldsymbol{\theta})$ is symmetric, it follows that $\partial \boldsymbol{\eta}/\partial \boldsymbol{\theta} = \mathbf{G}(\boldsymbol{\theta})$. □

Appendix F

Useful Results

F.1 Hoeffding Inequality

Theorem 16 (Hoeffding Inequality)**.** *Given a set $S = \{X_1, X_2, \ldots, X_n\}$ of independent random observations where $X_i \in [a_i, b_i]$. Then, let us denote $X = \frac{1}{n}\sum_{i=1}^{n} X_i$, the following inequality holds*

$$\Pr\left(|X - \mathbb{E}[X]| \geq t\right) \;\leq\; 2\exp\left(-\frac{2n^2t^2}{\sum_{i=1}^{n}(b_i - a_i)^2}\right) \;. \tag{F.1}$$

Proof Sketch. Omitted. □

F.2 Union Bound

Theorem 17 (Union Bound)**.** *Let $A_1, A_2, \ldots$ be a countable set of events, then*

$$\Pr\left[\bigcup_{i=1}^{\infty} A_i\right] \;\leq\; \sum_{i=1} \Pr\left(A_i\right) \;. \tag{F.2}$$

Proof Sketch. Omitted. □

F.3 Jensen Inequality

Theorem 18 (Jensen Inequality)**.** *Let X be a random variable. If f is a convex function then $\mathbb{E}[f(X)] \geq f(\mathbb{E}[X])$. Otherwise, if f is a concave function we have $\mathbb{E}[f(X)] \leq f(\mathbb{E}[X])$. For example, the exponential function $\exp(.)$ is convex while the logarithm function $\log(.)$ is concave.*

Proof Sketch. Omitted. □

F.4 Gershgorin Circle Theorem

Theorem 19 (Gershgorin Circle Theorem)**.** *Let A be a complex $n \times n$ matrix, with entries a_{ij}. For any row $1 \leq i \leq n$, let $R_i \stackrel{\triangle}{=} \sum_{i \neq j} |a_{ij}|$ denote the sum of the absolute values of the non-diagonal entries in the i^{th} row. Then, for any eigenvalue ψ of A, there exists i such that the following inequality holds:*

$$|\psi - a_{ii}| \leq R_i \,. \tag{F.3}$$

In case A is a real, positive definite matrix (e.g., the covariance matrix), the above inequality helps us bound A's smallest eigenvalue from below: $\psi_{\min} \geq a_{ii} - R_i$ for some i, which then implies $\psi_{\min} \geq \min_{i=1}^{n} (a_{ii} - R_i)$. In addition, if $a_{ii} = c$ is constant, we can further bound $\psi_{\min} \geq c - \max_{i=1}^{n} R_i$.

Proof Sketch. Omitted. □

F.5 Gaussian Tail Inequality

Theorem 20 (Gaussian Tail Inequality)**.** *Let X be a normal random variable: $X \sim \mathcal{N}(0,1)$. Then, we have*

$$\Pr(X > \epsilon) \;\leq\; \frac{1}{\epsilon}\exp\left(-\frac{\epsilon^2}{2}\right) \;. \tag{F.4}$$

By symmetry, we can further derive $\Pr(|X| > \epsilon) \leq \frac{2}{\epsilon}\exp\left(-\frac{\epsilon^2}{2}\right)$.

Proof Sketch. Omitted. □

Bibliography

[A.-Lopez *et al.*, 2012] M. A.-Lopez, V. Thomas, and O. Buffet. Near-optimal BRL using optimistic local transitions. In *Proc. ICML*, 2012. 6, 14, 17

[Akchurina, 2009] N. Akchurina. Multiagent reinforcement learning: Algorithm converging to Nash equilibrium in general-sum discounted stochastic games. In *Proc. AAMAS*, pages 725–732, 2009. 17

[Álvarez *et al.*, 2010] M. A. Álvarez, D. Luengo, M. K. Titsias, and N. D. Lawrence. Efficient multioutput gaussian processes through variational inducing kernels. In *Proc. AISTATS*, pages 25–32, 2010. 129

[Amari, 1998] S. Amari. Natural gradient works efficiently in learning. *Neural Computation*, 10:251–276, 1998. 93, 96, 97

[Anshelevich *et al.*, 2009] E. Anshelevich, D. Chakrabarty, A. Hate, and C. Swamy. Approximation algorithms for the firefighter problem: Cuts over time and submodularity. *Algorithms and Computation*, 2009. 8

[Asmuth and Littman, 2011] J. Asmuth and M. L. Littman. Learning is planning: Near Bayes-optimal reinforcement learning via Monte-Carlo tree search. In *Proc. UAI*, pages 19–26, 2011. 6, 14, 17

[Balcan *et al.*, 2009] M.-F. Balcan, A. Beygelzimer, and J. Langford. Agnostic active learning. *J. Comput. Syst. Sci.*, 75(1):78–89, 2009. 6, 21

[Beygelzimer *et al.*, 2009] A. Beygelzimer, S. Dasgupta, and J. Langford. Importance weighted active learning. In *Proc. NIPS*, 2009. 6, 21

[Bianchi *et al.*, 2007] R. A. C. Bianchi, C. H. C. Ribeiro, and A. H. R. Costa. Heuristic selection of actions in multiagent reinforcement learning. In *Proc. IJCAI*, 2007. 17

[Bishop, 2006] C. M. Bishop. *Pattern Recognition and Machine Learning*. Springer, 2006. 84

[Bonilla *et al.*, 2008] E. V. Bonilla, K. M. A. Chai, and C. K. I. Williams. Multi-task gaussian process prediction. In *Proc. NIPS*, 2008. 129

[Bowling and Veloso, 2001] M. Bowling and M. Veloso. Rational and convergent learning in stochastic games. In *Proc. IJCAI*, 2001. 17

[Boyle and Frean, 2005] P. Boyle and M. Frean. Dependent gaussian processes. In *Proc. NIPS*, 2005. 129

[Cao *et al.*, 2013] N. Cao, K. H. Low, and J. M. Dolan. Multi-robot informative path planning for active sensing of environmental phenomena: A tale of two algorithms. In *Proc. AAMAS*, pages 7–14, 2013. 6, 14, 19, 145

[Chalkiadakis and Boutilier, 2003] G. Chalkiadakis and C. Boutilier. Coordination in multiagent reinforcement learning: A Bayesian approach. In *Proc. AAMAS*, pages 709–716, 2003. 15, 18, 26, 37, 41, 43, 49

[Chen and Krause, 2013] Y. Chen and A. Krause. Near-optimal batch mode active learning and adaptive submodular optimization. In *Proc. ICML*, 2013. 8

[Chen *et al.*, 2012] J. Chen, K. H. Low, C. K.-Y. Tan, A. Oran, P. Jaillet, J. M. Dolan, and G. S. Sukhatme. Decentralized data fusion and active sensing with mobile sensors for modeling and predicting spatiotemporal traffic phenomena. In *Proc. UAI*, pages 163–173, 2012. 6, 14, 19, 78, 101

[Chen *et al.*, 2013a] J. Chen, N. Cao, K. H. Low, R. Ouyang, C. K.-Y. Tan, and P. Jaillet. Parallel gaussian process regression with low-rank covariance matrix approximations. In *Proc. UAI*, pages 152–161, 2013. 23

[Chen *et al.*, 2013b] J. Chen, N. Cao, K. H. Low, R. Ouyang, C. K.-Y. Tan, and P. Jaillet. Parallel Gaussian process regression with low-rank covariance matrix approximations. In *Proc. UAI*, pages 152–161, 2013. 23, 53, 100, 101

[Chen *et al.*, 2013c] J. Chen, K. H. Low, and C. K.-Y. Tan. Gaussian process-based decentralized data fusion and active sensing for mobility-on-demand system. In *Proc. RSS*, 2013. 19

[Cover and Thomas, 1991] T. Cover and J. Thomas. *Elements of Information Theory*. John Wiley & Sons, NY, 1991. 55, 69, 152, 163

[Cuong *et al.*, 2014] N. V. Cuong, W. S. Lee, and N. Ye. Near-optimal adaptive pool-based active learning with general loss. In *Proc. UAI*, 2014. 21

[Dasgupta *et al.*, 2007] S. Dasgupta, D. Hsu, and C. Monteleoni. A general agnostic active learning algorithm. In *Proc. NIPS*, 2007. 6, 21

[Dearden *et al.*, 1998] R. Dearden, N. Friedman, and S. Russell. Bayesian Q-learning. In *Proc. AAAI*, pages 761–768, 1998. 15, 41, 46

[Diggle, 2006] P. J. Diggle. Bayesian geostatistical design. *Scand. J. Statistics*, 33(1):53–64, 2006. 20, 52

[Dolan *et al.*, 2009] J. M. Dolan, G. Podnar, S. Stancliff, K. H. Low, A. Elfes, J. Higinbotham, J. C. Hosler, T. A. Moisan, and J. Moisan. Cooperative aquatic sensing using the telesupervised adaptive ocean sensor fleet. In *Proc. SPIE Conference on Remote Sensing of the Ocean, Sea Ice, and Large Water Regions*, volume 7473, 2009. 5, 19

[Duff, 2003] M. Duff. Design for an optimal probe. In *Proc. ICML*, 2003. 13, 30

[Gipps, 1981] P. G. Gipps. A behavioural car following model for computer simulation. *Transportation Research B*, 15(2):105–111, 1981. 15, 48

[Golovin and Krause, 2011] D. Golovin and A. Krause. Adaptive submodularity: Theory and applications in active learning and stochastic optimization. *Journal of Artificial Intelligence Research*, 42:427–486, 2011. 8

[Golovin *et al.*, 2010] D. Golovin, A. Krause, and D. Ray. Near-optimal Bayesian active learning with noisy observations. In *Proc. NIPS*, pages 766–774, 2010. 21

[Hanneke, 2007] S. Hanneke. A bound on the label complexity of agnostic active learning. In *Proc. ICML*, pages 353–360, 2007. 6, 21

[Hensman *et al.*, 2013] J. Hensman, N. Fusi, and N. D. Lawrence. Gaussian processes for big data. In *Proc. UAI*, 2013. 24, 88, 99, 100, 101

[Higdon, 2002] D. Higdon. Space and space-time modeling using process convolutions. *Journal of Quantitative Methods for Current Environmental Issues*, pages 37–56, 2002. 129

[Hoang and Low, 2012] T. N. Hoang and K. H. Low. Intention-aware planning under uncertainty for interacting with self-interested, boundedly rational agents. In *Proc. AAMAS*, pages 1233–1234, 2012. 15

[Hoang and Low, 2013a] T. N. Hoang and K. H. Low. A general framework for interacting Bayes-optimally with self-interested agents using arbitrary parametric model and model prior. In *Proc. IJCAI*, pages 1394–1400, 2013. 4, 5, 6, 15, 55, 125

[Hoang and Low, 2013b] T. N. Hoang and K. H. Low. Interactive POMDP Lite: Towards practical planning to predict and exploit intentions for interacting with self-interested agents. In *Proc. IJCAI*, 2013. 6, 15

[Hoang *et al.*, 2014] T. N. Hoang, K. H. Low, P. Jaillet, and M. Kankanhalli. Nonmyopic ϵ-Bayes-Optimal Active Learning of Gaussian Processes. In *Proc. ICML*, pages 739–747, 2014. 126

[Hoffman *et al.*, 2013] M. D. Hoffman, D. M. Blei, C. Wang, and J. Paisley. Stochastic variational inference. *Journal of Machine Learning Research*, pages 1303–1347, 2013. 93, 96

[Houlsby *et al.*, 2012] N. Houlsby, J. M. H.-Lobato, F. Huszar, and Z. Ghahramani. Collaborative Gaussian processes for preference learning. In *Proc. NIPS*, pages 2105–2113, 2012. 20, 52

[Huber *et al.*, 2008] M. Huber, T. Bailey, H. Durrant-Whyte, and U. D. Hanebeck. On entropy approximation for Gaussian mixture random vectors. In *Proc. IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems*, pages 181–188, 2008. 56

[Kemple *et al.*, 2003] D. Kemple, J. Kleinberg, and E. Tardos. Maximizing the spread of influence through a social network. In *Proc. KDD*, pages 137–146, 2003. 8

[Kolter and Ng, 2009] J. Z. Kolter and A. Y. Ng. Near-bayesian exploration in polynomial time. In *Proc. ICML*, 2009. 17

[Krause and Guestrin, 2007] A. Krause and C. Guestrin. Nonmyopic active learning of Gaussian processes: An exploration-exploitation approach. In *Proc. ICML*, pages 449–456, 2007. 5, 6, 11, 20, 22, 52, 53, 69, 74, 78, 150

[Krause *et al.*, 2008] A. Krause, A. Singh, and C. Guestrin. Near-optimal sensor placements in Gaussian processes: Theory, efficient algorithms and empirical studies. *JMLR*, 9:235–284, 2008. 19

[Kurniawati *et al.*, 2008] H. Kurniawati, D. Hsu, and W. S. Lee. Sarsop: Efficient point-based pomdp planning by approximating optimally reachable belief spaces. In *Proc. Robotics: Science and Systems*, 2008. 16, 43

[Lázaro-Gredilla *et al.*, 2010] M. Lázaro-Gredilla, J. Quiñonero-Candela, C. E. Rasmussen, and A. R. Figueiras-Vidal. Sparse spectrum gaussian process regression. *Journal of Machine Learning Research*, pages 1865–1881, 2010. 23

[Leonard *et al.*, 2007] N. E. Leonard, D. A. Palley, F. Lekien, R. Sepulchre, D. M. Fratantoni, and R. E. Davis. Collective motion, sensor networks, and ocean sampling. *Proc. IEEE*, 95(1):48–74, 2007. 5, 19

[Li *et al.*, 2009] L. Li, J. D. Williams, and S. Balakrishnan. Reinforcement learning for spoken dialog management using least-squares policy iteration and fast feature selection. In *Proceedings of the 10th Annual Conference of the International Speech Communication Association*, pages 2475–2478, 2009. 4

[Li *et al.*, 2010] L. Li, W. Chu, J. Langford, and R. E. Schapire. A contextual-bandit approach to personalized news article recommendation. In *Proceedings of the 19th International Conference on World Wide Web*, pages 661–670, 2010. 4

[Littman, 2001] Michael L. Littman. Friend-or-foe Q-learning in general-sum games. In *Proc. ICML*, 2001. 17

[Low *et al.*, 2007] K. H. Low, G. J. Gordon, J. M. Dolan, and P. Khosla. Adaptive sampling for multi-robot wide-area exploration. In *Proc. IEEE ICRA*, pages 755–760, 2007. 5, 14, 18

[Low *et al.*, 2008] K. H. Low, J. M. Dolan, and P. Khosla. Adaptive multi-robot wide-area exploration and mapping. In *Proc. AAMAS*, pages 23–30, 2008. 4, 14, 19

[Low *et al.*, 2009] K. H. Low, J. M. Dolan, and P. Khosla. Information-theoretic approach to efficient adaptive path planning for mobile robotic environmental sensing. In *Proc. ICAPS*, pages 233–240, 2009. 4, 14, 19, 56

[Low *et al.*, 2011] K. H. Low, J. M. Dolan, and P. Khosla. Active Markov information-theoretic path planning for robotic environmental sensing. In *Proc. AAMAS*, pages 753–760, 2011. 6, 14, 19

[Low *et al.*, 2012] K. H. Low, J. Chen, J. M. Dolan, S. Chien, and D. R. Thompson. Decentralized active robotic exploration and mapping for probabilistic field classification in environmental sensing. In *Proc. AAMAS*, pages 105–112, 2012. 6, 14, 19

[Martin, 2001] R. J. Martin. Comparing and contrasting some environmental and experimental design problems. *Environmetrics*, 12(3):303–317, 2001. 4, 20

[Müller, 2007] W. G. Müller. *Collecting Spatial Data: Optimum Design of Experiments for Random Fields*. Springer, 3rd edition, 2007. 4, 20

[Natarajan *et al.*, 2012a] P. Natarajan, T. N. Hoang, K. H. Low, and M. Kankanhalli. Decision-theoretic approach to maximizing observation of multiple targets in multi-camera surveillance. In *Proc. AAMAS*, pages 155–162, 2012. 6, 14

[Natarajan *et al.*, 2012b] P. Natarajan, T. N. Hoang, K. H. Low, and M. Kankanhalli. Decision-theoretic coordination and control for active multi-camera surveillance in uncertain, partially observable environments. In *Proc. ICDSC*, 2012. 6, 14

[Natarajan *et al.*, 2014] P. Natarajan, T. N. Hoang, Y. Wong, K. H. Low, and M. Kankanhalli. Scalable decision-theoretic coordination and control for real-time active multi-camera surveillance. In *Proc. ICDSC*, 2014. 14

[Ouyang *et al.*, 2014] R. Ouyang, K. H. Low, J. Chen, and P. Jaillet. Multi-robot active sensing of non-stationary Gaussian process-based environmental phenomena. In *Proc. AAMAS*, 2014. 20, 52

[Park and Pillow, 2012] M. Park and J. W. Pillow. Bayesian active learning with localized priors for fast receptive field characterization. In *Proc. NIPS*, pages 2357–2365, 2012. 20, 52

[Pineau *et al.*, 2003] J. Pineau, G. Gordon, and S. Thrun. Point-based value iteration: An anytime algorithm for POMDPs. In *Proc. IJCAI*, pages 1025–1032, 2003. 16, 34, 40

[Podnar *et al.*, 2010] G. Podnar, J. M. Dolan, K. H. Low, and A. Elfes. Telesupervised remote surface water quality sensing. In *Proc. IEEE Aerospace Conference*, 2010. 5, 19

[Poupart and Vlassis, 2008] P/ Poupart and N. Vlassis. Model-based bayesian reinforcement learning in partially observable domains. In *Proc. ISAIM*, 2008. 13

[Poupart *et al.*, 2006] P. Poupart, N. Vlassis, J. Hoey, and K. Regan. An analytic solution to discrete Bayesian reinforcement learning. In *Proc. ICML*, pages 697–704, 2006. 3, 6, 7, 13, 15, 28, 29, 30, 32, 33, 41, 42, 43, 44, 45, 46, 55, 125

[Powers and Shoham, 2005] R. Powers and Y. Shoham. Learning against opponents with bounded memory. In *Proc. IJCAI*, 2005. 18

[Quiñonero-Candela and Rasmussen, 2005] J. Quiñonero-Candela and C. E. Rasmussen. A unifying view of sparse approximate gaussian process regression. *Journal of Machine Learning Research*, 6:1939–1959, 2005. 23, 80, 82, 83, 94, 95, 99, 104, 178, 181, 182, 183, 187, 190

[Rasmussen and Williams, 2006] C. E. Rasmussen and C. K. I. Williams. *Gaussian Processes for Machine Learning*. MIT Press, 2006. 22, 53, 81, 82, 84, 89, 90, 102, 179, 180, 187

[Riedmiller *et al.*, 2009] M. Riedmiller, T. Gabel, R. Hafner, and S. Lange. Reinforcement learning for robot soccer. *Automaton Robot*, 27:55–73, 2009. 4

[Robbins and Monro, 1951] H. Robbins and S. Monro. A stochastic approximation method. In *The Annals of Mathematical Statistics*, pages 400–407, 1951. 86, 92

[Ross *et al.*, 2007] S. Ross, B. Chaib-draa, and J. Pineau. Bayes-adaptive pomdps. In *Proc. NIPS*, 2007. 13

[Rue and Held, 2005] H. Rue and L. Held. *Gaussian Markov Random Fields: Theory and Applications*. Chapman & Hall/CRC, 2005. 61

[Sabou *et al.*, 2014] M. Sabou, K. Bontchenva, L. Derczynski, and A. Scharl. Corpus annotation through crowdsourcing: Towards best practice guidelines. In *Proc. LREC*, pages 859–866, 2014. 8

[Schwaighofer and Tresp, 2003] A. Schwaighofer and V. Tresp. Transductive and inductive methods for approximate gaussian process regression. In *Proc. NIPS*, pages 953–960, 2003. 23

[Seeger *et al.*, 2003] M. Seeger, C. K. I. Williams, and N. D. Lawrence. Fast forward selection to speed up sparse gaussian process regression. In *Proceedings of the 9th International Workshop on AISTATS*, 2003. 23, 86, 99, 104, 183

[Shewry and Wynn, 1987] M. C. Shewry and H. P. Wynn. Maximum entropy sampling. *J. Applied Statistics*, 14(2):165–170, 1987. 56, 73

[Singh *et al.*, 2009] A. Singh, A. Krause, C. Guestrin, and W. J. Kaiser. Efficient informative sensing using multiple robots. *J. Artificial Intelligence Research*, 34:707–755, 2009. 19

[Smola and Bartlett, 2001] A. J. Smola and P. L. Bartlett. Sparse greedy gaussian process regression. In *Proc. NIPS*, pages 619–625, 2001. 23, 183, 190

[Snelson and Ghahramani, 2007] E. L. Snelson and Z. Ghahramani. Local and global sparse Gaussian process approximation. In *Proc. AISTATS*, 2007. 23, 24, 82, 83, 94, 95, 105, 178, 181, 182, 183, 185

[Snelson and Gharahmani, 2006] E. Snelson and Z. Gharahmani. Sparse gaussian processes using pseudo-inputs. In *Proc. NIPS*, pages 1259–1266, 2006. 23

[Snelson, 2007] E. L. Snelson. *Flexible and efficient Gaussian process models for machine learning*. PhD thesis, Gatsby Computational Neuroscience Unit, University of London, 17 Queen Square, London WC1N 3AR, United Kingdom, 2007. 24, 80, 99, 104

[Solomon and Zacks, 1970] H. Solomon and S. Zacks. Optimal design of sampling from finite populations: A critical review and indication of new research areas. *J. American Statistical Association*, 65(330):653–677, 1970. 7, 11, 20, 52

[Soong, 2004] T. T. Soong. *Fundamentals of Probability and Statistics for Engineers*. John Wiley & Sons, 2004. 57

[Sorg *et al.*, 2010] J. Sorg, S. Singh, and R. L. Lewis. Variance-based rewards for approximate bayesian reinforcement learning. In *Proc. UAI*, 2010. 17

[Spaan and Vlassis, 2005] M. T. J. Spaan and N. Vlassis. Perseus: Randomized point-based value iteration for POMDPs. *Journal of Artificial Intelligence Research*, 2005. 16

[Stone *et al.*, 2005] P. Stone, R. S. Sutton, and G. Kuhlmann. Reinforcement learning for robocup soccer keepaway. *International Society for Adaptive Behavior*, 13(3):165–188, 2005. 4

[Suematsu and Hayashi, 2002] N. Suematsu and A. Hayashi. A multiagent reinforcement learning algorithm using extended optimal response. In *Proc. AAMAS*, 2002. 17

[Tesauro, 2003] G. Tesauro. Extending Q-learning to general adaptive multi-agent systems. In *Proc. NIPS*, 2003. 17

[Titsias, 2009] M. K. Titsias. Variational learning of inducing variables in sparse gaussian processes. In *Proceedings of the 12th International Workshop on AISTATS*, 2009. 23, 24, 82, 83, 84, 85, 86, 88

[Tokekar *et al.*, 2013] P. Tokekar, J. V. Hook, D. Mulla, and V. Isler. Sensor planning for a symbiotic UAV and UGV system for precision agriculture. In *Proc. IROS*, 2013. 5, 18

[Vijayakumar *et al.*, 2005] S. Vijayakumar, A. D'Souza, and S. Schaal. Incremental online learning in high dimensions. *Neural Computation*, 17(12):2602–2634, 2005. 100

[Wang *et al.*, 2012] Y. Wang, K. S. Won, D. Hsu, and W. S. Lee. Monte carlo bayesian reinforcement learning. In *Proc. ICML*, 2012. 4, 16, 26, 37, 38, 40, 41, 43, 44, 46, 47, 49

[Williams, 2006] J. D. Williams. *Partially Observable Markov Decision Processes for Spoken Dialogue Management*. PhD thesis, Cambridge University, 2006. 4

[Yang *et al.*, 2011] L. Yang, S. Hanneke, and J. Carbonell. The sample complexity of self-verifying Bayesian active learning. In *Proc. AISTATS*, pages 816–822, 2011. 6, 21

[Zimmerman, 2006] D. L. Zimmerman. Optimal network design for spatial prediction, covariance parameter estimation, and empirical prediction. *Environmetrics*, 17(6):635–652, 2006. 20, 52