ADAPTIVE SENSING IN UNCERTAIN ENVIRONMENTS: MAXIMUM LIKELIHOOD, SENSOR NETWORKS, AND REINFORCEMENT LEARNING

by

Doron Blatt

A dissertation submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
(Electrical Engineering: Systems)
in The University of Michigan
2006

Doctoral Committee:
Professor Alfred O. Hero III, Chair
Professor Jeffrey A. Fessler
Professor Susan Murphy
Professor David L. Neuhoff
Associate Professor Satinder Singh Baveja

To Ruth.

ACKNOWLEDGEMENTS

First and foremost I would like to thank my advisor, Professor Alfred Hero, for his guidance, inspiration, mentorship, and friendship. I admire Professor Hero's curiosity and love for science, his charisma, and his leadership with his students and colleagues, and I am grateful for the opportunity to work under his supervision. I will carry the lessons I have learned from our interactions for the years to come.

I would also like to thank the members of my dissertation committee, Professor Satinder Baveja, Professor Jeffrey Fessler, Professor Susan Murphy, and Professor David Neuhoff, for their careful reading of the dissertation and helpful comments.

Special thanks go to Professor Susan Murphy. Our numerous interactions in classes, seminars, reading groups, and one-on-one meetings gave rise to many of the ideas that appear in the dissertation. I feel fortunate for the opportunity to meet a scholar of her caliber and learn from her unique way of analyzing problems. I am also grateful for her openness and friendship.

I was lucky to have my research interests intersect with Professor Fessler's and would like to thank him and Dr. Sangtae Ahn for their collaboration.

I would like to thank my wife, Ruth Blatt, for her unconditional support throughout the years and for carrying Tamar for nine months, thus relieving me of that burden. I would also like to thank my family in Israel for believing in me and for their support.

Finally, I would like to thank my friends for being there for me and for helpful

discussions about research and life in general. In particular, my thanks go to Daniel Almiral, Jose Costa, Daniel Marco, Neal Patwari, Raviv Raich, and Ji Zhu.

The research presented in this thesis was partially funded by NIH/NCI grant 1P01 CA87634, by DARPA-MURI grant ARO DAAD 19-02-1-0262, by NSF contract CCR-0325571, and by fellowship of the Department of Electrical Engineering and Computer Science, University of Michigan.

TABLE OF CONTENTS

LIST OF FIGURES

Figure

ABSTRACT

The advent of distributed and agile sensing systems that collect data in multiple locations and through a variety of sensing modalities has brought about new and exciting challenges to the field of signal processing. Motivated by problems that arise in the development of these systems, the thesis makes contributions in three domains: (1) distributed optimization for inference in sensor networks, (2) statistical tests for optimality that mitigate the problem of sensitivity to local maxima, and (3) development and analysis of reinforcement learning solutions to stochastic decision problems for resource allocation in agile sensing.

A novel incremental gradient method, called incremental aggregated gradient (IAG), that can be used by wireless sensor networks to perform inference in a distributed manner, is proposed and analyzed. A gradient aggregation concept relaxes the common requirement of incremental methods for a diminishing step size for convergence, and a fast convergence rate is established.

The convergence of IAG is established under a certain unimodality assumption. For non-convex problems however, for example when IAG is applied to find the maximum likelihood estimator, the method might stagnate at a local maximum. To mitigate this weakness, the following question is addressed: Given the location of a relative maximum of the log-likelihood function, how to assess whether it is the global maximum? We analyze and improve an existing statistical tool, called A Test for Global Maximum, that answers this question by posing it as a hypothesis

testing problem. Tests that are insensitive to model mismatch are proposed, thereby overcoming a fundamental weakness of this tool.

Finding optimal policies for controlling an agile sensing system is formulated as a reinforcement learning problem and solved via a novel approximate dynamic programming algorithm that approximates the solution of the associated multi-stage non-convex optimization problem by solving a sequence of single-stage convex problems. Via this approximation a plethora of off-the-shelf classification methods can be applied to approximate the solution of the more complicated reinforcement learning problem. The consequences of the approximation are investigated by deriving finite sample upper bounds on the performance of the estimated policy relative to the performance of the optimal one.

CHAPTER 1

Introduction

1.1 Overview

The advent of distributed and agile sensing systems that collect data in multiple locations and through a variety of sensing modalities has brought about new and exciting challenges to the field of signal processing. This phrase "distributed and agile sensing systems" ties together the emerging technologies of wireless sensor networks [35,69] and multi-modal sensing systems [60,68]. A wireless sensor network is a system of partially connected data acquisition elements that are spatially distributed to sample a random field. Wireless sensor networks were introduced to accomplish monitoring tasks such as measuring power consumption over the electric power grid to prevent overloads, collecting traffic volumes over the internet to identify abnormalities, environmental monitoring, and surveillance. Agile, or multi-modal, sensing is the capability of controlling the data collection process. Examples of agile sensing systems include a radar that can control its beam direction, a land mine detector that can deploy several types of sensors, and a monitoring satellite that can control the frequency band of its radar. The key element that differentiates agile sensing systems from other data collection systems is a resource allocation constraint that

precludes using all sensor modalities at all times. In its operation, an agile sensing system must select the best sensing modality based on past observations to maximize a given objective. These two technologies are clearly linked in the following way. If, due to power and bandwidth constraints, a sensor network must activate only a portion of its elements, then the resulting system can be analyzed as an agile sensing system. If an agile sensing system is composed of several platforms that are connected though noisy channels, then we encounter similar design problems as we do in sensor networks.

The contribution of the thesis to the development of distributed and agile sensing systems will be described as we review various aspects of an intrusion detection system. Consider a wireless network of acoustic sensors that is distributed to monitor a field. When a vehicle enters the field, the network should track its position. Given the sensors' measurements, it is easy to derive an estimator of the source location. However, the fact that wireless sensors are powered by batteries limits the amount of information that can be transmitted by the sensors, and hence, a central design problem is how to compute this estimator from data that are distributed across the network elements. Solutions to this problem divide into centralized and distributed methods (see Fig. 1.1). In the centralized approach, the data collected by the sensors are communicated to a central point (sometimes called a fusion center) for processing [117]. Distributed methods relax the requirement for complete data sharing. They solve the optimization problem via iterative partial information sharing between the network elements. Distributed methods are especially advantageous when the data collected by the sensors are only the means for performing inference, as in the application described above. In this case, transmitting the data from each sensor to a fusion center may be unnecessary [90]. Several contributions have been made to distributed optimization for sensor networks. In Chapter 2 a novel incremental

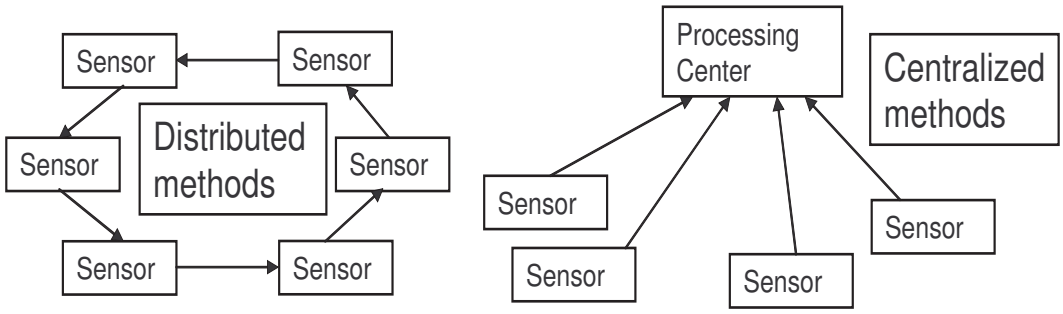

Figure 1.1: Centralized and Distributed methods.

gradient method, called incremental aggregated gradient (IAG), that can be used by wireless sensor networks to perform inference in a distributed manner, is proposed and analyzed. As explained in more details in Section 1.2 below, a gradient aggregation concept relaxes the common requirement of incremental methods for a diminishing step size for convergence [18, 48, 51, 64, 76, 77, 79, 108]. Sensor network acoustic source localization, which was traditionally solved via the maximum likelihood estimator [75, 94, 95, 105], is formulated as a convex feasibility problem in [J1] and solved via a distributed version of the projection onto convex sets (POCS) algorithm. In [C2], the gradient aggregation approach of Chapter 2 is applied to derive a convergent variation of POCS. The gradient aggregation approach was also applied outside of the domain of wireless sensor networks. In [J2] the approach is applied to derive a convergent incremental optimization transfer algorithm for tomography.

In Chapter 2, convergence of IAG is established under a certain unimodality assumption. For non-convex problems however, for example when IAG is applied to find the maximum likelihood estimator of the source location, the method might stagnate at a local maximum. The sensitivity to local maxima is common to other methods for computing the maximum likelihood estimator, such as the expectation maximization algorithm [38] or Fisher scoring [81]. To mitigate this weakness, the following question is addressed: Given a location of a relative maximum of the log-

likelihood function, how to assess whether this is the global maximum? As detailed in Section 1.3, Chapter 3 analyzes an existing statistical tool [21,39,49], called A Test for Global Maximum, that answers this question by posing it as a hypothesis testing problem. Furthermore, tests that are insensitive to model mismatch are proposed, thereby overcoming a fundamental weakness of this tool.

As part of the analysis in Chapter 3, the asymptotic characterization of the local maxima of the likelihood function is derived. This result was used to propose a method for solving the global optimization problem of maximum likelihood in wireless sensor networks. Suppose that each sensor finds a local maximum of the likelihood function and transmits its location, rather than the collected data, to the fusion center. It is shown in [C3] that the problem of finding the maximum likelihood estimator from the sub-optimal local estimates is a Gaussian mixture clustering problem, and that it is possible to identify the cluster that is associated with estimates that converged to the global maximum by comparing the covariance of the clusters to the inverse of the Fisher Information Matrix. Once the right cluster is identified, its mean is the final estimator.

A common approach for saving battery power in wireless sensor networks is to adaptively switch the sensors on and off, in accordance with the task and the state of the network. For example, in the intrusion detection system described above, one might activate a small portion of the sensors at first, and once an initial estimate of the source location is obtained, more sensors in the vicinity of the source are turned on while distant sensors are switched off. A closely related problem is that of finding policies to control an agile sensing system [60,68], which we formulate as a sequential choice of experiment problem [37]. In the sequential choice of experiment problem, a system performs inference based on information that can be acquired through a number of sensors or sensor modalities, each with a different observation distribution

and deployment cost (see Fig. 1.2). The goal is to find a policy for controlling the agile sensing system that achieves optimal inference performance with a minimal number of sensor dwells. In this thesis, we consider the more general problem of finding optimal policies for controlling an arbitrary finite horizon stochastic decision process, which includes the sequential choice of experiment problem as a special case, and treat the problem of finding the optimal policy without explicit knowledge of the underlying model, but rather from experimental or simulated data. This model free instance of the stochastic control problem is called reinforcement learning [112]. In [J3], the reinforcement learning algorithm Q-learning [112] was applied to find near-optimal policies for a multi-modal radar to detect maneuvering targets. It is well known, however, that applying reinforcement learning methods to real life applications requires a great deal of expertise and experimentation [102]. In [C1], a Gauss-Seidel algorithm is used to break a multistage reinforcement learning problem into a sequence of single-stage reinforcement learning subproblems, which are then converted to supervised learning problems that can be solved using off-the-shelf methods. The goal was to leverage techniques and theoretical results from supervised learning for solving the more complex problem of reinforcement learning, as advocated in [9]. In Chapter 4, we build on the ideas in [C1] and propose an approximate dynamic programming algorithm to solve the reinforcement learning problem. The algorithm approximates the multi-stage non-convex optimization problem, required for finding the optimal policy, by a sequence of single-stage convex problems. Via this approximation a plethora of off-the-shelf classification methods can be applied to approximate the solution of the more complicated reinforcement learning problem. The consequences of the approximation are investigated by deriving finite sample upper bounds on the performance of the estimated policy relative to the optimal one. In Chapter 5 we return to the sequential choice of experiment problem and the

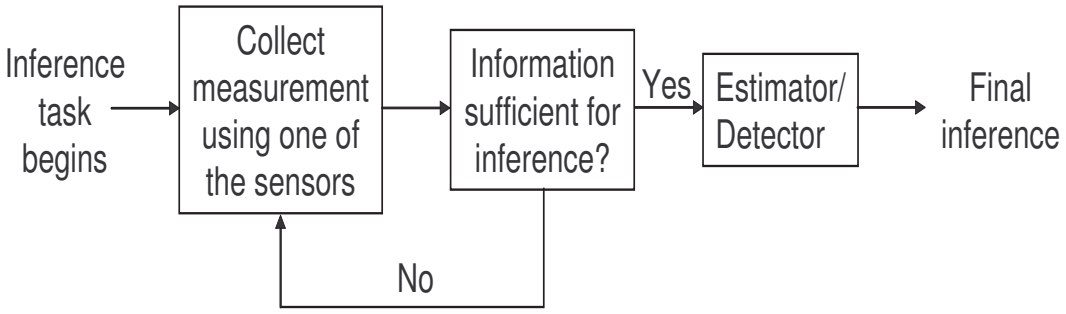

Figure 1.2: Agile sensing system.

approximate dynamic programming algorithm is successfully applied to the problems of finding an optimal policy for controlling a land mine detection system from simulated data and finding an optimal policy for controlling a land monitoring satellite from real data.

In Figure 1.3, the different elements of the presented work and the relations between them is presented.

The thesis is organized as follows. This chapter reviews the results in the thesis: Sections 1.2, 1.3, and 1.4 summarize the results on the IAG method, the tests for global maximum, and the work on reinforcement learning, respectively. In Chapters 2 and 3 two self-contained manuscripts based on the results on the IAG method and the tests for global maximum are given. In Chapter 4 the approximate dynamic algorithm and its analysis are presented. The application of the algorithm to the sequential choice of experiment problem is presented in Chapter 5.

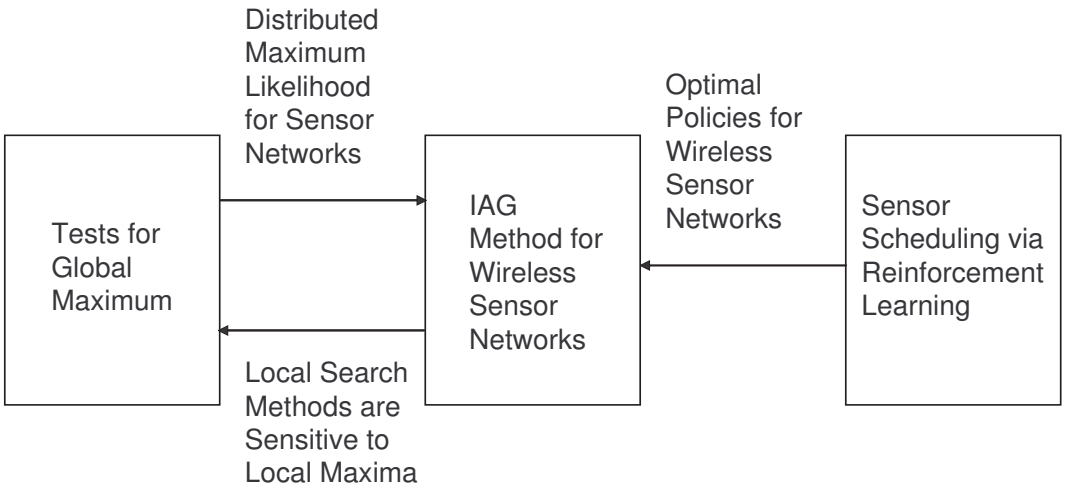

Figure 1.3: Structure of Thesis.

Major Publications

Journal Papers

[J1] D. Blatt and A. O. Hero, "Energy Based Sensor Network Source Localization via Projection onto Convex Sets (POCS)", to appear in IEEE Trans. on Signal Processing, 2005.

[J2] S. Ahn, J. A. Fessler, D. Blatt, and A. O. Hero, "Convergent incremental optimization transfer algorithms: application to tomography", IEEE Trans. on Medical Imaging, Volume 25, Issue 3, March 2006, Pages:283-296.

[J3] C. Kreucher, D. Blatt, A. O. Hero, and K. Kastella, "Adaptive multi-modality sensor scheduling for detection and tracking of smart targets", In press, Digital Signal Processing, 2006.

[J4] D. Blatt and A. O. Hero, "On tests for global maximum of the log-likelihood function", accepted after revision at IEEE Trans. on Information Theory, 2005.

[J5] D. Blatt, A. O. Hero, and H. Gauchman, "A convergent incremental gradient algorithm with a constant stepsize", conditional acceptance SIAM Journal on Optimization, 2006.

Conference Papers

[C1] D. Blatt and A. O. Hero, "From Weighted Classification to Policy Search", to appear in Proceedings of the Nineteenth Annual Conference on Neural Information Processing Systems (NIPS), 2006.

[C2] D. Blatt and A. O. Hero, "APOCS: A Rapidly Convergent Source Localization Algorithm for Sensor Networks", IEEE Workshop on Statistical Signal Processing (SSP), Bordeaux, July 2005.

[C3] D. Blatt and A. O. Hero, "Distributed Maximum Likelihood for Sensor Networks", Proceedings of the 2004 IEEE International Conference on Acoustics, Speech, and Signal Processing, Montreal, Canada, May 2004.

1.2 Distributed Optimization for Sensor Networks

This section describes the results presented in a paper that is under second review at *SIAM Journal on Optimization*.

1.2.1 Introduction

The emerging technology of wireless sensor networks provides means of efficiently collecting data that are distributed over a large area [69]. The collection of the data, however, is often not the end goal but rather the means for solving an associated optimization problem, e.g., inference [90]. Solutions to the optimization problems that arise in sensor networks divide into centralized and distributed methods. In the centralized approach, the data collected by the sensors are communicated to a central point (sometimes called a fusion center) for processing. Distributed methods relax the requirement for complete data sharing by solving the optimization problem via iterative partial information sharing between the network elements. A comparison between the centralized and distributed frameworks shows that, when the monitored area is fixed, transmitting data to a fusion center requires the transmission of $O(L)$ bits over a distance of $O(1)$ per bit, whereas sharing information in a distributed manner requires $O(L)$ bits over a distance of $O(\sqrt{\log^2 L/L})$ per bit, where L is the number of sensors [96]. Hence, distributed implementation is advantageous for dense networks. Furthermore, when the data collected by the sensors are only the means for performing inference, transmitting the data from each sensor to a fusion center may be unnecessary and distributed schemes can be used to perform the same inference task in a distributed manner while reducing the communication and energy requirements [90]. In particular, under the assumption that data collected at different

sensors are independent, inference optimization problems often take the form

$$\text{minimize} \qquad f(x) = \sum_{l=1}^{L} f_l(x), \quad x \in \mathbb{R}^p, \tag{1.1}$$

where $f_l : \mathbb{R}^p \to \mathbb{R}$ are indexed by the data at sensor l, $l = 1, \ldots, L$. For example, in maximum likelihood estimation $f_l(x)$ is the log-likelihood of the measurements of sensor l given the parameter vector x, and in least squares estimation $f_l(x)$ corresponds to the sum of squared distances between the measurements of sensor l to the assumed parametric model parameterized by x.

To solve (1.1) in a distributed manner, Rabbat and Nowak [95] proposed the incremental gradient algorithm that has been studied extensively in the literature [18, 48,51,52,64,76,77,79,108]. To describe the incremental gradient method, the steepest descent method is first reviewed. Given an initial point x^1, the steepest descent method generates a sequence $\{x^k\}_{k\geq 1}$ according to

$$x^{k+1} = x^k - \mu \nabla f(x^{x^k}) = x^k - \mu \sum_{l=1}^{L} \nabla f_l(x^k),$$

where μ is a positive constant step size chosen small enough to ensure convergence. Hence, every update requires the computation of each of the gradients $\nabla f_l(x^k)$, $l = 1, \ldots, L$. In contrast, the incremental gradient method updates x^k according to

$$x^{k+1} = x^k - \mu(k) \nabla f_{(k)_L}(x^k), \tag{1.2}$$

where $\mu(k)$ is a positive step size, possibly depending on k, and $(k)_L$ denotes k modulo L. Hence, the incremental gradient method requires a single gradient evaluation per iteration. When this algorithm is implemented in a sensor network, (1.1) is solved via several communication cycles across the network. Sensor 1 generates x^1 arbitrar-

ily, computes x^2 according to (1.2) and transmits x^2 to sensor 2. Upon receiving x^k and from sensor $(k-1)_L$, processor $(k)_L$ computes x^{k+1} according to (1.2) and transmits the new estimate x^{k+1} to sensor $(k+1)_L$. Incremental gradient methods are motivated by the observation that when the iterates are far from the eventual limit, the evaluation of a single gradient component is sufficient for generating an approximate descent direction. Hence, these methods lead to a significant reduction in the amount of required computations per iteration (see e.g. [16] section 1.5.2 and the discussion in [15]). When implemented in a sensor network, these methods relax the requirement for transmitting the data from all the sensors to a fusion center [95]. The drawback of these methods, when using a constant step size, is that the iterates converge to a limit cycle and oscillate around a stationary point [76], unless restrictions of the type $\nabla f_l(x) = 0$, $l = 1, \ldots, L$ whenever $\nabla f(x) = 0$ are imposed [108]. In a sensor network application, this amounts to having each sensor converge to a different limit. Convergence for a diminishing step size has been established by a number of authors under different conditions [18, 48, 51, 64, 76, 77, 79, 108]. However, a diminishing step size usually leads to slow convergence near the eventual limit and requires exhaustive experimentation to determine how rapidly the step size must decrease in order to prevent scenarios in which the step size becomes too small when the iterates are far from the eventual limit (e.g. determining the constants a and b in step sizes of the form $\mu(k) = a/(k+b)$).

Chapter 2 proposes and analyzes a novel incremental gradient method called *incremental aggregated gradient* (IAG) for solving (1.1), which requires a single gradient computation per iteration and converges with a constant step size. The IAG method generates a sequence $\{x^k\}_{k\geq 1}$ as follows. Given L arbitrary initial points $x^1, x^2, \ldots, x^L$, an aggregated gradient, denoted by d^L, is defined as $\sum_{l=1}^{L} \nabla f_l(x^l)$.

For $k \geq L$,

$$x^{k+1} = x^k - \mu \frac{1}{L} d^k, \tag{1.3}$$

$$d^{k+1} = d^k - \nabla f_{(k+1)_L}(x^{k+1-L}) + \nabla f_{(k+1)_L}(x^{k+1}), \tag{1.4}$$

where μ is a positive constant step size chosen small enough to ensure convergence, and the factor $1/L$ is explicitly included to make the approximate descent direction $\frac{1}{L}d^k$ comparable in magnitude to the one used in the standard incremental gradient method (1.2). Thus, at every iteration a new point x^{k+1} is generated according to the direction of the aggregated gradient d^k. Then, only one of the gradient summands $\nabla f_{(k+1)_L}(x^{k+1})$ is computed to replace the previously computed $\nabla f_{(k+1)_L}(x^{k+1-L})$. Note that for $k \geq L$ the IAG iteration (1.3)–(1.4) is equivalent to

$$x^{k+1} = x^k - \mu \frac{1}{L} \sum_{l=0}^{L-1} \nabla f_{(k-l)_L}(x^{k-l}). \tag{1.5}$$

It is seen that the principal difference between the standard incremental gradient method (1.2) and the IAG method is that the standard incremental gradient method uses only one of the components in order to generate an approximate descent direction, whereas the IAG method uses the average of the L previously computed gradients. This property leads to convergence of the IAG method for fixed and sufficiently small positive step size μ. This is as contrasted to the standard incremental gradient method, whose convergence requires that the step size sequence $\mu(k)$ converge to zero.

A hybrid between the steepest descent method and the incremental gradient method was studied in [15]. The hybrid method starts as an incremental gradient method and gradually becomes the steepest descent. This method requires a tuning parameter, which controls the transition between the two methods, to gradually

increase with k to ensure convergence. When the tuning parameter increases sufficiently fast with the number of iterations, it is shown that the rate of convergence is linear. However, the question of determining the rate of transition between the two methods still remains. For any fixed value of the tuning parameter, the hybrid method converges to a limit cycle, unless a diminishing step size is used, similar to the standard incremental gradient method.

The choice of the aggregated gradient d^k (1.4) for generating an approximate descent direction was mentioned in [51] in the context of adaptive step size methods, which require repeated evaluations of either the complete objective function $f(x)$ or its gradient. This requirement renders the methods proposed in [51] inapplicable to problems in sensor networks of interest to us or any other applications which require decentralized implementation. In addition, as noted in [116], if $\nabla f_l(x)$, $l = 1, \ldots, L$, are not necessarily zero whenever $\nabla f(x) = 0$, the step size tends to zero, resulting in slow convergence.

The IAG method is closely related to Tseng's incremental gradient with momentum term [116], which is an incremental generalization of Polyak's heavy-ball method [91, p. 65] (also called the steepest descent with momentum term [17, p. 104]). Rewriting Tseng's method's update rule as

$$x^{k+1} = x^k - \mu(k) \sum_{l=0}^{k} \zeta^l \nabla f_{(k-l)_L}(x^{k-l}),$$

we see from (1.5) that the IAG method is a variation of this method with a truncated sum, $\zeta = 1$, and a constant step size. Similar to [51], the step size adaptation rule that leads to convergence in [116] requires repeated evaluations of the complete objective function $f(x)$ and its gradient. Hence, this method cannot be implemented in a distributed manner either. Furthermore, a linear convergence rate is established

only under a certain growth property on the functions' gradients, which requires $\nabla f_l(x) = 0$, $l = 1, \ldots, L$, whenever $\nabla f(x) = 0$.

In contrast to the available methods, the IAG method has all four of the following properties: (a) it evaluates a single gradient per iteration, (b) it uses a constant step size, (c) it is convergent, and (d) it has global linear convergence rate for quadratic objective $f(x)$.

1.2.2 Convergence Analysis

The convergence analysis is done for several function classes. Under the assumption that $\nabla f_l(x)$, $l = 1, \ldots, L$ are bounded and satisfy a Lipschitz condition it is shown that

$$\liminf_{k\to\infty} ||\nabla f(x^k)|| \leq \frac{C_1}{1 - C_2}\mu,$$

where C_1 and C_2 are given constants. To prove this result the IAG method is first written in a form known as the "gradient method with errors" [18]:

$$\begin{aligned} x^{k+1} &= x^k - \mu\frac{1}{L}\left[\sum_{l=0}^{L-1} \nabla f_{(k-l)_L}(x^k) + \sum_{l=0}^{L-1} \nabla f_{(k-l)_L}(x^{k-l}) - \sum_{l=0}^{L-1} \nabla f_{(k-l)_L}(x^k)\right] \\ &= x^k - \mu\frac{1}{L}\left[\nabla f(x^k) + h^k\right], \end{aligned} \tag{1.6}$$

where

$$h^k = \sum_{l=1}^{L-1} \left[\nabla f_{(k-l)_L}(x^{k-l}) - \nabla f_{(k-l)_L}(x^k)\right]$$

is the error term in the calculation of the gradient at x^k. Then it is shown that the process $\{h^k\}_{k\geq 1}$ can be upper bounded by the output of an autoregressive linear system that is driven by a bounded process. Then a standard analysis of a gradient method with *bounded* errors leads to the result.

Next, it is assumed in addition that $f(x)$ has a unique global minimum at x^*,

that the Hessian of $f(x)$ is continuous and positive definite at x^*, and that for any sequence $\{t^k\}_{k=1}^{\infty}$ in $\mathbb{R}^p$, if $\lim_{k\to\infty} f(t^k) = f(x^*)$ or $\lim_{k\to\infty} ||\nabla f(t^k)|| = 0$, then $\lim_{k\to\infty} t^k = x^*$. Under these assumptions, pointwise convergence of the method is established. The proof follows the following argument. As long as the the gradient of $f(x)$ at x^k is larger than the maximal contribution of the error term in (1.6), d^k is a descent direction and the iterates reduce the value of the function. When the iterates enter a region of small gradient, the method slows down, i.e., the distance between subsequent iterations reduces, and a new bound on the contribution of the error term h^k can be established. The convergence result follows by iteratively applying these two arguments.

It is shown that the assumptions required for the convergence result are weaker than strict convexity, which is usually assumed to establish global convergence. In particular, two examples of functions that satisfy all the assumptions are given. These are an objective function associated with a robust estimator [55] and the objective function associated with the LogitBoost algorithm [46]. The case when $f(x)$ and $f_l(x)$ are quadratic functions, however, is not covered by the above analysis. For this important case, a completely different convergence proof is given and it is shown in addition that the convergence rate is globally linear.

In the quadratic case, the functions f_l, $l = 1, \ldots, L$, have the following form

$$f_l(x) = \frac{1}{2} x' Q_l x - c_l' x, \quad l = 1, \ldots, L, \tag{1.7}$$

where Q_l are given symmetric matrices, c_l are given vectors, and $\sum_{l=1}^{L} Q_l$ is positive definite. Under this assumption, the function $f(x) = \sum_{l=1}^{L} f_l(x)$ is strictly convex, has its minimum point at

$$x^* = (\sum_{l=1}^{L} Q_l)^{-1} \sum_{l=1}^{L} c_l, \tag{1.8}$$

and x^* is the only stationary point of $f(x)$. In Chapter 2, it is shown that for sufficiently small μ, $\lim_{k\to\infty} x^k = x^*$ and the rate of convergence of the IAG method is linear. To prove the convergence result in this case, the iterates are first written explicitly

$$x^{k+1} = x^k - \mu \left[\sum_{l=0}^{L-1} Q_{(k-l)_L} x^{k-l} - c_{(k-l)_L} \right] = x^k - \mu \sum_{l=0}^{L-1} Q_{(k-l)_L} x^{k-l} + \mu c,$$

where $c = \sum_{l=1}^{L} c_l$, and the factor $\frac{1}{L}$ was absorbed into μ to simplify the notation. Subtracting x^* (1.8) from both sides and adding and subtracting x^* inside the parentheses, we obtain

$$x^{k+1} - x^* = x^k - x^* - \mu \sum_{l=0}^{L-1} Q_{(k-l)_L} (x^{k-l} - x^* + x^*) + \mu c.$$

Denoting the error at the kth iteration by $e^k = x^k - x^*$ and the substitution of (1.8) for x^* lead to the following error form

$$e^{k+1} = e^k - \mu \sum_{l=0}^{L-1} Q_{(k-l)_L} e^{k-l}.$$

This relation between a new error and the previous errors can be seen as a periodically time varying linear system. To analyze its stability, which will lead to the convergence result, it is useful to consider L iterations as one iteration [82]. This can be seen as down-sampling the original system by a factor of L, which leads to a time invariant

system of a lower sampling rate. Specifically, by defining

$$\overline{e}^k = \begin{bmatrix} e^k \\ e^{k-1} \\ \vdots \\ e^{k-L+1} \end{bmatrix},$$

it is shown by induction that

$$\overline{e}^{k+L} = M(\mu)\overline{e}^k,$$

where $M(\mu)$ is a matrix function of μ. Therefore, to establish convergence (and a linear rate), we need to prove that the eigenvalues of $M(\mu)$ are inside the unit circle for sufficiently small μ. It is easy to see that if $\mu = 0$, $M(\mu)$ has multiple eigenvalues at zero and one. By continuity, the eigenvalue at zero will remain inside the unit circle for small enough μ. As for the eigenvalues at one, it is shown that they enter the unit circle as μ increases from zero by showing that the derivative of the function that expresses the dependency of the eigenvalues on μ is negative at $\mu = 0^+$.

1.2.3 Application to Sensor Networks

In Chapter 2, it is shown how the IAG method is implemented in a sensor network in a distributed manner, i.e, applied to solve optimization problems without sending the data to a fusion center. For two sensor network applications, numerical experiments compare the IAG method with other incremental gradient methods, showing the advantages of the new method.

1.3 Tests for Global Maximum of the Likelihood Function

This section describes the results presented in a paper that has been accepted after revision to the *IEEE Transactions on Information Theory*.

1.3.1 Introduction

Chapter 3 tackles a question that is fundamental to Maximum Likelihood estimation: Given a location of a relative maximum of the likelihood function, how to assess whether this is the global maximum? The problem of distinguishing between local and global maxima arises whenever the Maximum Likelihood method is applied to nonlinear problems and local search methods, such as the Expectation Maximization algorithm [38], Fisher scoring [81], or the IAG method of Chapter 2, are applied. In Chapter 3, a statistical tool, called A Test for Global Maximum [21, 39, 49], that answers this question by posing it as a hypothesis testing problem is analyzed. The analysis quantifies the sensitivity of the tests to model mismatch in terms of the Renyi divergence and the Kullback-Leibler distance between the true underlying distribution and the assumed parametric class. The analysis also leads to a simple threshold correction method that accounts for possible deviations from the model as long as these deviations are bounded in terms of the mentioned distances. When deviations from the model are defined in terms of an embedding in a larger parametric class, insensitivity to a Pitman drift is established by constructing tests based on a vector valued validation function that is orthogonal to the elements of the gradient of the log-likelihood function of the larger class. This construction leads to tests that are locally robust to deviations from the assumed model. Finally, the tests are applied to three problems that are known to suffer from local maxima: (1) passive localization

using an array of sensors, (2) clustering by estimating the parameters of a Gaussian mixture model, and (3) time series analysis though estimation of superimposed exponentials in noise.

1.3.2 Preliminaries

Let y_t, $t = 1, \ldots, n$ be a collection of n independent observations drawn from an unknown distribution G with density $g(y)$, $y \in \mathbb{R}^P$. The information we want to extract from the data is encoded in a $K \times 1$ parameter vector θ, through which we define a parametric family of densities $\{f(y, \theta) : \theta \in \Theta\}$. Denote by

$$L_n(Y_n; \theta) = \frac{1}{n} \sum_{t=1}^{n} \log f(y_t; \theta)$$

the normalized log-likelihood function of the measurements, where $Y_n = [y_1\, y_2\, \ldots\, y_n]$. The MLE[1] is defined as

$$\widehat{\theta}_n = \arg \max_{\theta \in \Theta} L_n(Y_n; \theta). \tag{1.9}$$

Denote by $\mathrm{E}\{\cdot\}$ the expectation with respect to the true underlying distribution G, and by θ^* the minimizer of the Kullback-Leibler information, i.e.,

$$\theta^* = \arg \min_{\theta \in \Theta} \mathrm{E}\left\{ \log \frac{g(y)}{f(y; \theta)} \right\} = \arg \max_{\theta \in \Theta} a(\theta)$$

where $a(\theta)$ is the ambiguity function, defined as

$$a(\theta) = \mathrm{E}\{\log f(y; \theta)\} \tag{1.10}$$

and assume that θ^* is unique.

[1] Sometimes called quasi-MLE when the model is incorrect.

Denote by $\widetilde{\theta}_n$ one of the relative maxima of the log-likelihood function. Then the problem addressed in this paper can be formulated as a hypothesis testing problem. Given $\widetilde{\theta}_n$, decide between

$$\begin{aligned} \mathrm{H}_0 : &\quad \widetilde{\theta}_n = \widehat{\theta}_n \\ \mathrm{H}_1 : &\quad \widetilde{\theta}_n \neq \widehat{\theta}_n. \end{aligned} \tag{1.11}$$

A statistical test which gives a solution to this problem is called a *test for global maximum*.

M-Tests for Global Maximum

M-tests were proposed in an econometric context by Newey [88], Tauchen [114], and White [123] as a general way of testing the validity of parametric models (see [124, Ch. 9] as well). The tests are based on a vector valued test function

$$e(y, \theta) : \mathbb{R}^P \times \Theta \to \mathbb{R}^Q \tag{1.12}$$

which is chosen to satisfy

$$\int e(y, \theta) f(y, \theta) dy = 0, \quad \forall \theta \in \Theta. \tag{1.13}$$

Hence, given the MLE $\widehat{\theta}_n$, large values (relative to a threshold to be defined below) of $1/n \sum_{t=1}^{n} e(y_t, \widehat{\theta}_n)$ indicate that a model mismatch is likely. Small values of $1/n \sum_{t=1}^{n} e(y_t, \widehat{\theta}_n)$ indicate that the model is correctly specified or alternatively that the type of model mismatch is such that $g(y) \notin \{f(y, \theta) : \theta \in \Theta\}$ but

$$\mathrm{E}\left\{e(y, \theta^*)\right\} = \int e(y, \theta^*) g(y) dy = 0. \tag{1.14}$$

The same framework can be used to construct tests for (1.11). First suppose that the model is correctly specified and that $e(y, \theta)$ is chosen to satisfy (1.13). Then, given a location of a relative maximum of the log-likelihood function $\widetilde{\theta}_n$, large values of $1/n \sum_{t=1}^{n} e(y_t, \widetilde{\theta}_n)$ indicate that it is not likely that $\widetilde{\theta}_n$ is the MLE. This directly extends to the case of model mismatch, if it is known that (1.14) holds.

To constructed a test for global maximum choose a function $e(y, \theta)$ that satisfies (1.14). The function $e(y, \theta)$ will be called the *global maximum validation function.* Define the vector

$$h_n(\theta) = \frac{1}{n} \sum_{t=1}^{n} e(y_t, \theta) \tag{1.15}$$

and assume that $V_n(\widehat{\theta}_n)$ is a given consistent estimator for the asymptotic covariance matrix of $h_n(\widehat{\theta}_n)$. In Chapter 3, several possibilities for such a consistent estimator that are available in the literature are reviewed. It is possible to show that under H_0 the statistic

$$S_n = n h_n^T(\widetilde{\theta}_n) V_n^{-1}(\widetilde{\theta}_n) h_n(\widetilde{\theta}_n) \tag{1.16}$$

is asymptotically Chi-Squared distributed with Q degrees of freedom, denoted by χ_Q^2. Denote by $F_{\chi_Q^2}(\cdot)$ the χ_Q^2 cumulative distribution function. Therefore, a false alarm level α test of the hypotheses (1.11) is made by comparing S_n to $F_{\chi_Q^2}^{-1}(1-\alpha)$, which is the critical value of the χ_Q^2 distribution for the desired false alarm level. If S_n exceeds the critical value, H_0 is rejected and one concludes that the iterative local search should be re-initiated in the hope of convergence to a different maximum. Otherwise, the null hypothesis cannot be rejected and $\widetilde{\theta}_n$ is declared the final estimate.

In Chapter 3 it is shown that two tests for global maximum that are available in the literature fall into this framework and that it is easy to construct other tests, e.g., from global maximum validation functions that are based on the moments of

the underlying distribution.

1.3.3 Power Analysis

The power function of a test is the probability of correctly rejecting the null hypothesis as a function of the specified level. In order to derive the power function, the asymptotic distribution of $\widetilde{\theta}_n$ under H_1 needs to be determined. Therefore, assumptions on the structure of the ambiguity function (1.10) at different local maxima are required. Assume that the system of equations $\nabla a(\theta) = 0$, has a finite number of solutions in Θ and each one of these solutions is an interior point of Θ. In addition, at each of these points, the matrix $\nabla^2 a(\theta)$ is either negative definite or positive definite. The ambiguity function $a(\theta)$ has its global maximum at θ^*; denote by θ^m, $m = 1, \ldots, M$, the other M local maxima of $a(\theta)$. In Chapter 3 it is proven that for sufficiently large n, $L_n(Y_n; \theta)$ has $M + 1$ local maxima for almost every sequence $\{y_t\}_{t \geq 1}$ and that the location of these relative maxima are strongly consistent estimates for θ^* and θ^m, $m = 1, \ldots, M$. This result ensures that as n increases the relative maxima of the log-likelihood function occur close to the relative maxima of the ambiguity function and only at these locations.

Based on this result, it is shown that the test statistic S_n (1.16) is asymptotically distributed as a non-central χ^2 random variable and the consistency of the tests is established, i.e., it is shown that the power function converges to one as n increases to infinity for every choice of level $\alpha \in (0, 1)$. In addition, the non-centrality parameter of the non-central χ^2 distribution is derived and, based on it, a finite n approximation to the power function is given. This result extends the results of [49] and [21], which established under a correctly specified model (each for their own global maximum

validation function) that if the only solution to the set of equations

$$\int \nabla_\theta \log f(y,\theta) f(y,\theta^0) dy = 0$$
$$\int e(y,\theta) f(y,\theta^0) dy = 0$$

is θ^0, then

$$\sqrt{n} h_n(\widetilde{\theta}_n) \xrightarrow{D} N(0, V(\theta^0)) \quad \text{iff} \quad \widetilde{\theta}_n = \widehat{\theta}_n.$$

1.3.4 Misspecified Models

When $g(y) \in \{f(y,\theta) : \theta \in \Theta\}$ we say that the model is correctly specified. When there exists no θ for which $g(y) = f(y,\theta)$ we say that the model is misspecified. In general, it is difficult to discriminate between the cases of: (a) $\widetilde{\theta}_n$ a local maximum in a correctly specified model; and (b) $\widetilde{\theta}_n$ a global maximum in a misspecified model. Under model mismatch, the probability of mistakenly rejecting $\widetilde{\theta}_n$ as the global maximum, increases with the number of samples.

If the test statistic is designed under the assumption that the model is correctly specified but the actual underlying distribution $g(y)$ is outside the assumed parametric family $\{f(y,\theta) : \theta \in \Theta\}$, then (1.14) may be violated. In this case, even when $\widetilde{\theta}_n = \widehat{\theta}_n$, $h_n(\widetilde{\theta}_n) \xrightarrow{a.s.} \mathrm{E}\{e(y,\theta^*)\} = h(\theta^*) \neq 0$ and, similar to the discussion in the previous section, S_n is approximately distributed as non-central χ^2, instead of the assumed central chi-squared. In this case, as n tends to infinity, the probability of mistakenly rejecting $\widetilde{\theta}_n$ as the global maximum increases to one regardless of the test threshold. In Chapter 3 two ways to overcome this weakness are given.

A Bound on the Non-Centrality Parameter

It is possible to bound the non-centrality parameter that is associated with the model mismatch in terms of the Renyi divergence between $f(y;\theta^*)$ and true underlying density $g(y)$. Furthermore, if the true underlying distribution is restricted to a larger parametric class which contains the assumed model as a subspace, then a bound in terms of the Kullback-Leibler distance can be easily computed. Based on the bounds, a threshold correction method is given, which leads to tests that are robust to small deviations from the model as long as these deviations are bounded in terms of the mentioned distances. For example, in array signal processing it is often assumed for simplicity that the noise terms at the array antenna elements are uncorrelated. In practice, however, the antennas are mounted on the same platform and electro-magnetic interference cause correlation between the noise terms. In Chapter 3 it is shown that if the noise terms are modelled as a spacial autoregressive process then it is possible to bound the effect of this model mismatch on the test and to correct the test's threshold accordingly so as to obtain a robust test.

Tests Insensitive to a Pitman Drift

Assume that the parametric class $\{f(y;\theta) : \theta \in \Theta\}$ is embedded in a larger class $\{\widetilde{f}(y;\theta,\gamma) : \theta \in \Theta, \gamma \in \Gamma \subset \mathbb{R}^{K'}\}$ such that $f(y;\theta) = \widetilde{f}(y;\theta,\gamma^0)$ for all $\theta \in \Theta$. Furthermore, assume that the true underlying distribution depends on n, hence denoted by $g_n(y)$, and is given by

$$g_n(y) = \widetilde{f}(y;\theta^0,\gamma^0 + \gamma/\sqrt{n}) \tag{1.17}$$

for some fixed $\gamma \in \Gamma$, and denote the limiting distribution by $g(y)$. In the context of model specification tests, this type of local alternative is called a Pitman drift.

Newey [88] investigated the performance of M-tests in this scenario and used the result to maximize the power of the tests against such local alternatives. Here, our goal is reversed; we would like the tests to be insensitive to small deviations from the assumed model. By considering the space of zero-mean L_2 functions of y with inner product

$$\langle f_1(y), f_2(y) \rangle = \int f_1(y) f_2(y) f(y;\theta) dy$$

it is shown that our objective is to construct a global maximum validation function $e(y,\theta)$, with elements orthogonal to the space spanned by the set of functions

$$\nabla_\beta \log \widetilde{f}(y;\beta)\Big|_{\gamma=\gamma^0}, \tag{1.18}$$

where $\beta = [\theta^T, \gamma^T]^T$ is the concatenated parameter vector. Given any global maximum validation function $e(y,\theta)$, it is shown in Chapter 3 how to construct the component $e^\perp(y,\theta)$, which is orthogonal to the functions in (1.18). By this construction, we obtain a test which is insensitive to the Pitman drift regardless of the vector γ. Denoting the classes of log-likelihood functions $\{\log f(y;\theta) : \theta \in \Theta\}$ and $\{\log \widetilde{f}(y;\theta,\gamma) : \theta \in \Theta,\ \gamma \in \Gamma\}$ by $\mathcal{F}$ and $\mathcal{G}$, respectively, Fig. 1.4 gives a geometrical interpretation of the construction of $e^\perp(y,\theta)$. For the array signal processing mismatch model example, it is shown that this construction indeed leads to tests that are robust to small deviations from the assumed model.

1.3.5 Applications

The asymptotic regime adopted in the analysis, raises the question of small sample performance. In Chapter 3, tests for global maximum are derived and evaluated through simulations for several parameter estimation problems. In the simulations the following aspects were studied. First, the accuracy of setting the test threshold

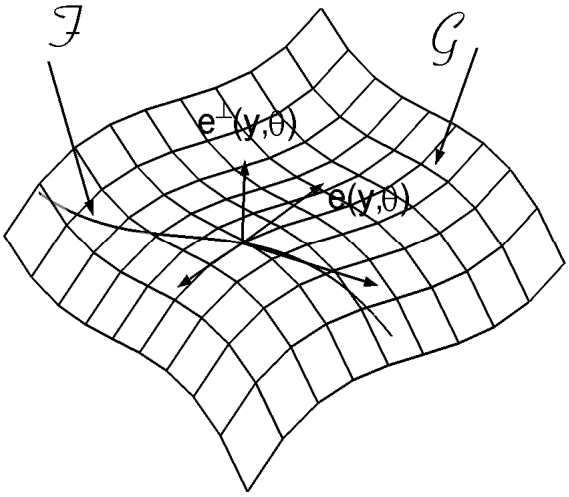

Figure 1.4: Geometrical interpretation of the construction of tests insensitive to Pitman drift.

to $F^{-1}_{\chi^2_Q}(1-\alpha)$ for a level α test was evaluated. Second, we evaluated how fast the power of the test approaches 1, as the number of samples increases, and the accuracy of the finite sample power approximation. Finally, the sensitivity of the tests to a misspecified model is examined and the threshold adjustment procedure and the construction of tests that are orthogonal to deviations from the model are demonstrated.

1.4 Reinforcement Learning

This section describes results that were presented at the eighteenth annual conference on neural information processing systems (NIPS) 2005 and more recent results that have not been published yet. These results are given in Chapters 4 and 5. As mentioned earlier the research is motivated by the sequential choice of experiment problem that arise in agile sensing systems. However, the algorithm and analysis are more general and apply to the larger class of finite horizon stochastic decision processes, which contain the sequential choice of experiment problem as a special case. Hence, in Chapter 4 we consider the finite horizon stochastic control problem, and in particular, its model free case – the reinforcement learning problem. Then, in Chapter 5, we apply the results to the special case of the sequential choice of experiment problem.

1.4.1 Introduction

The field of reinforcement learning is centered around the challenge of designing agents that learn to act in a stochastic environment by interacting with it [112]. As the agent interacts with the environment it receives rewards, and the goal is to eventually learn through these rewards which actions maximize the future sum of rewards. There are a number of mathematical model for reinforcement learning. In this thesis we treat the problem of finding the optimal policy for controlling a finite horizon partially observable stochastic decision process. Such a process consists of several elements:

- **The decision epochs** determine the times in which the agent must take an action. In the discrete model adopted here, decision epochs occur at $t = 0, 1, \ldots, T$, where we consider the case in which T is finite.

- At every decision epoch, prior to taking an action, the agent collects an **observation** of the system's state, denoted $O_t \in \mathcal{O}_t$, $t = 0, 1, \ldots, T$. In general, O_t is a combination of observable system state variables and noisy measurements of partially observable system variables, and, in general, contains discrete-valued and continuous-valued elements.

- At every decision epoch the agent chooses an **action** A_t, based on the previous observations, from a set of possible actions called the **action space** $\mathcal{A}$. We assume that $\mathcal{A}$ is a finite set. When T is finite, the action at time T is the final action.

- Upon taking action A_t at time t, the agent observes the next observation O_{t+1} and receives a reward, denoted by $r(\mathbf{O}_t, \mathbf{A}_t, O_{t+1})$, whose value depends on the past observations $\mathbf{O}_t = (O_0, O_1, \ldots, O_t)$ and actions $\mathbf{A}_t = (A_0, A_1, \ldots, A_t)$, and on the new observation O_{t+1}.

- A **deterministic policy** π is a sequence of mappings from all possible observations and actions histories to $\mathcal{A}$, which specifies the action to take at each decision epoch, given the history. When T is finite, the policy is composed of $T + 1$ mappings $\pi = (\pi_0, \pi_1, \ldots, \pi_T)$.

- A **random policy** $\pi = (\pi_{p_0}, \pi_{p_1}, \ldots, \pi_{p_T})$ is a sequence of conditional distribution functions over the action space $\mathcal{A}$ given all possible observations and actions histories. That is, $p_t(\cdot|\mathbf{o}_t, \mathbf{a}_{t-1})$ is a distribution over $\mathcal{A}$ for any realization $(\mathbf{o}_t, \mathbf{a}_{t-1})$ of the past observations and actions. When the system is controlled using a random policy, the actions are chosen at random according to these conditional distributions.

- A policy induces a distribution over the vector $O_0, A_0, O_1, \ldots, O_T, A_T, O_{T+1}$.

One common objective in stochastic control is to find the policy that maximizes the expected sum of rewards:

$$V(\pi) = \mathrm{E}_\pi \left\{ \sum_{t=0}^{T} r(\mathbf{O}_t, \mathbf{A}_t, O_{t+1}) \right\}, \tag{1.19}$$

where the expectation is taken with respect to the distribution induced by the policy π, hence the symbol E is subscript by π. The expected sum of rewards under policy π, $V(\pi)$, is called the averaged value function of the policy π. The optimal policy π^* is the policy that maximizes $V(\pi)$.

1.4.2 The Generative Model Assumption

A center problem in reinforcement learning is to find a policy that maximizes (1.19) by merely observing the controlled system, without knowledge of the transition probabilities. For example, under the generative model assumption [62] the initial distribution of O_0 and the distribution of O_t given past observations and actions are unknown but it is possible to generate realizations of the initial observations and generate a realization of O_t conditioned on arbitrary observations and actions histories. Chapter 4 considers the problem of estimating the optimal policy for controlling a finite horizon stochastic decision process based on n trajectory trees generated by a generative model. Each trajectory tree is generated as follows: The root of the tree is a random realization of O_0. Given the realization of the initial state, realizations of the next observation O_1 given all possible actions, denoted by O_1^a, $a \in \mathcal{A}$, are randomly generated. Each of the realizations of O_1 is now the root of the subtree. Denote by $O_t^{\mathbf{a}_{t-1}}$, where $\mathbf{a}_t = (a_0, a_1, \ldots, a_t)$, the random variable generated at the node that follows the sequence of actions $a_0, a_1, \ldots, a_{t-1}$. This random variable is a realization of O_t conditioned on the sequence of actions and the sequence of observa-

tions that appear on the path of the tree that leads to it. These iterations continue to generate a depth $T+1$ tree (See Fig. 1.5).

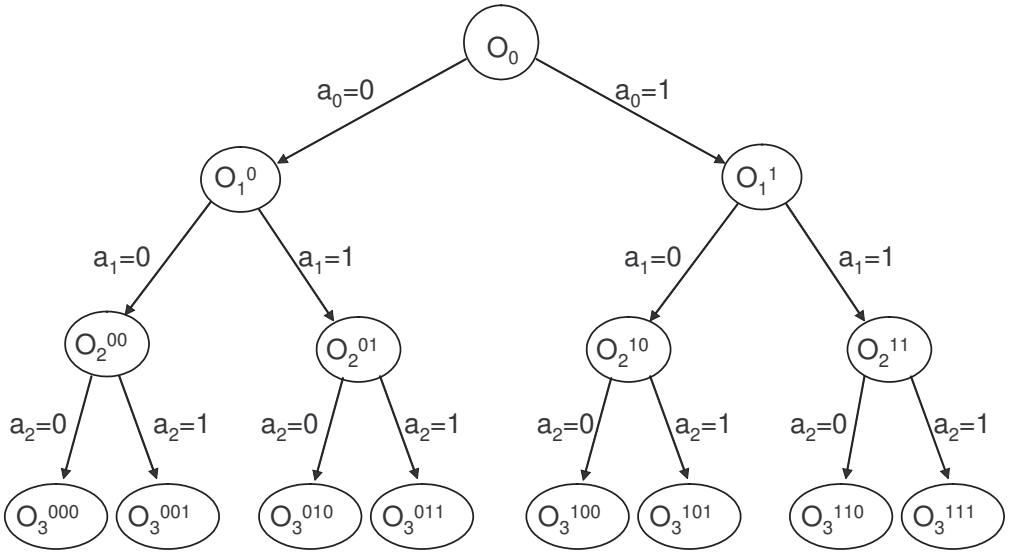

Figure 1.5: A binary trajectory tree of depth $T+1=3$.

Consider a class of deterministic policies Π, i.e., each element of Π is a sequence of $T+1$ mappings from histories to $\mathcal{A}$. It is possible to estimate the average value function value of any policy in the class from the set of trajectory trees by averaging the collection of the sum of rewards along the path of actions that agrees with the policy on each tree [62]. A policy specifies the action to take at each decision epoch and so there is exactly one path on every tree that agrees with a given policy. Denote by $\widehat{V}^i(\pi)$ the observed sum of rewards on the i'th tree along the path that corresponds to the policy π. Then the value of the policy π is estimated by

$$\widehat{V}_n(\pi) = n^{-1} \sum_{i=1}^{n} \widehat{V}^i(\pi). \tag{1.20}$$

In [62], the authors show that with high probability (over the data set) $\widehat{V}_n(\pi)$ converges uniformly over the policy class to $V(\pi)$ with rates that depend on the VC-

dimension [5] of the policy class. This result motivates the use of policies π with high $\widehat{V}_n(\pi)$, since with high probability these policies have high values of $V(\pi)$.

1.4.3 An Approximate Dynamic Programming Algorithm

In Chapter 4, we consider the problem of estimating the optimal policy from a restricted class of policies of the form $\Pi = \{(\pi_0, \pi_1, \ldots, \pi_T) : \pi_0 \in \Pi_0, \pi_1 \in \Pi_1, \ldots, \pi_T \in \Pi_T\}$. It is shown that while the task of finding the global optimum within a class of non-stationary policies may be overwhelming, an approximate dynamic programming algorithm leads to a sequence of single-stage reinforcement learning subproblems, which can be reduced to a sequence of weighted classification problems. Thus the algorithm converts a reinforcement learning problem into simpler supervised learning subproblems and the implication is that a plethora of classification methods can be applied to find policies in the reinforcement learning problem, which will enable tackling more complicated reinforcement learning problems in large state spaces using existing methods. The proposed algorithm first estimates the optimal policy for time T by considering the single-stage reinforcement learning problem following a randomly selected leaf at stage T. Given $\widehat{\pi}_T$, the actions following time T that do not agree with $\widehat{\pi}_T$ are removed from the tree. To find $\widehat{\pi}_{T-1}$ given $\widehat{\pi}_T$, a random leaf is selected on every tree at stage $T-1$. Then $\widehat{\pi}_{T-1}$ is the solution to the single-stage reinforcement learning problem for the sum of the two rewards following time $T-1$. Note that due to the tree pruning the reward at time T are deterministic functions of the previous history. This procedure continuous until the root of the tree. To solve each of the single-stage reinforcement learning problems, we propose a reduction to a weighted classification problems which leads the way to a convex approximation to a combinatorial optimization problem. Together with this reduction, the approximate dynamic programming algorithm is a link between

the reinforcement learning problem and existing tools for solving the simper classification problem. In other words, a plethora of off-the-shelf classification method can be applied to approximate the solution to the reinforcement learning problem.

1.4.4 Generalization Error Bounds

To support the proposed algorithm, we derive finite sample upper bounds of the type derived in [84] on the generalization error of the resulting estimated policy. The approach we take is similar to the one in [84]. Namely, we first write the generalization error in terms whose empirical counterparts are minimized by the algorithm, and then invoke uniform convergence results to bound these terms. However, the rates we establish are faster than the one in [84], except for the case in which the approximation class is a linear space, for which we establish the same rates. The rate improvement supports the use of weighted classification based methods over Q-learning methods that are based on regression. The bounds provide estimates for the number of trajectory trees required to achieve a given performance guarantee. But more importantly, the bounds establish a link between algorithms and analysis tools from the supervised learning literature and the reinforcement learning problem.

1.4.5 Application to the Sequential Choice of Experiment Problem

In Chapter 5, we return to the sequential choice of experiment problem that arises in sensor scheduling application and formulate it as a finite horizon partially observable stochastic decision problem. When formulated as a sequential choice[2]

[2]The key difference from the related sequential design of experiment problem is that instead of adapting a set of continuous experiment parameters, here we choose from a finite set of fixed experiments.

of experiments problem [37], the agile sensing problem consists of an episodic task that is divided into a sequence of decision epochs. Each episode begins as the first observation is collected. Then, at each subsequent decision epoch two decisions are made. The first one is to decide if the amount of information collected thus far is sufficient for making inference (detection or estimation) on the data with a desired accuracy or whether more observations are required. This first decision also determines the choices available at the second decision. If more observations are required, the next best sensor modality needs to be determined. If the information is deemed sufficient for inference, the final estimation or detection decision is made. Every sensor modality has an associated deployment cost and a decision rule must balance the expected information gain from a sensor deployment, which results in improved inference capabilities, with the deployment cost. The collection of decision rules, i.e., the sequence of mappings from past observations to the decision space, is called a policy and the goal is to find a policy that optimally balances the overall average sensor deployment costs and the estimation or detection performance, e.g., mean squared estimation error or classification error rate.

In the sequential choice of experiment problem we model the sensors outputs as a collection $X_1, X_2, \ldots, X_K$ of K random variables. Let Y be a discrete random variable that represent the state of nature whose value we try to predict. A policy π specifies which sensor to deploy first, say sensor k. Then, based of the value of X_k, the policy determines if an accurate prediction of Y is possible, and if so, what is the best prediction, or, otherwise, which is the next best sensor to deploy to collect additional data. This process continues until either a prediction of Y is made or all available sensors are deployed. We assume that each sensor can be applied at most once and hence, the total number observations is bounded by K. Therefore, a policy π is sequence of $K+1$ decision rules $\pi = [\pi_1, \pi_2, \ldots, \pi_{K+1}]$.

When represented as a finite horizon partially observable stochastic decision process, the initial observation O_0 is a constant that indicated that an episode begins but contains no information regarding the state of nature, Y. The first decision rule specifies the best sensor to deploy first. The result of deploying a sensor is an observation O_1 which corresponds to the sensor's output. The reward for taking this action is minus the deployment cost associated with the chosen sensor. The sensor selection process continues until it is decided that the information that was collected by the sequence of sensors is sufficient for predicting Y, and so the last decision of every episode is a prediction $\widehat{Y}$. The reward following this action is one if $\widehat{Y} = Y$ and zero otherwise.

We assume that deploying a sensor does not effect Y and so it is possible to populate an entire trajectory tree from a single realization of the random variables $X_1, X_2, \ldots, X_K, Y$. In particular, given this realization, it is possible to construct the path specified by any policy. In general, only a subset of the observation random variables and Y will appear on the path, unless it so happen that, for the specific realization of the observation random variables, the policy specifies taking all possible observations prior to making a prediction of Y. Therefore, the approximate dynamic programming algorithm of Chapter 4 can be directly applied to estimate the optimal policy for the sequential design of experiment problem based on a set of realizations of the random variables $X_1, X_2, \ldots, X_K, Y$.

In Chapter 5, we apply the approximate dynamic programming algorithm to estimate the optimal policy for controlling a land mine detection system from simulated data and finding an optimal policy for controlling a land monitoring satellite from real data.

CHAPTER 2

A Convergent Incremental Gradient Method with a Constant Step Size

2.1 Introduction

Consider the unconstrained optimization problem

$$\text{minimize} \qquad f(x) = \sum_{l=1}^{L} f_l(x), \quad x \in \mathbb{R}^p, \tag{2.1}$$

where $\mathbb{R}^p$ is the p-dimensional Euclidean space, and $f_l : \mathbb{R}^p \to \mathbb{R}$ are continuously differentiable scalar functions on $\mathbb{R}^p$. Our interest in this problem stems from optimization problems arising in wireless sensor networks (see e.g. [24, 90, 94–96]), in which $f_l(x)$ corresponds to the data collected by the lth sensor in the network. This problem also arises in neural network training, in which $f_l(x)$ corresponds to the lth training data set (see e.g. [17, 48, 51, 77–79]).

The iterative method proposed and analyzed in this paper for solving (2.1), which we call the *incremental aggregated gradient* (IAG) method, generates a sequence $\{x^k\}_{k\geq 1}$ as follows. Given L arbitrary initial points $x^1, x^2, \ldots, x^L$, an aggregated gradient, denoted by d^L, is defined as $\sum_{l=1}^{L} \nabla f_l(x^l)$. Possible initializations are dis-

cussed in §3. For $k \geq L$,

$$x^{k+1} = x^k - \mu \frac{1}{L} d^k, \tag{2.2}$$

$$d^{k+1} = d^k - \nabla f_{(k+1)_L}(x^{k+1-L}) + \nabla f_{(k+1)_L}(x^{k+1}), \tag{2.3}$$

where μ is a positive constant step size chosen small enough to ensure convergence, $(k)_L$ denotes k modulo L with representative class $\{1, 2, \ldots, L\}$, and the factor $1/L$ is explicitly included to make the approximate descent direction $\frac{1}{L} d^k$ comparable in magnitude to the one used in the standard incremental gradient method to be discussed below. Thus, at every iteration a new point x^{k+1} is generated according to the direction of the aggregated gradient d^k. Then, only one of the gradient summands $\nabla f_{(k+1)_L}(x^{k+1})$ is computed to replace the previously computed $\nabla f_{(k+1)_L}(x^{k+1-L})$. Note that for $k \geq L$ the IAG iteration (2.2)–(2.3) is equivalent to

$$x^{k+1} = x^k - \mu \frac{1}{L} \sum_{l=0}^{L-1} \nabla f_{(k-l)_L}(x^{k-l}). \tag{2.4}$$

The IAG method is related to the large class of incremental gradient methods that has been studied extensively in the literature [18, 48, 51, 52, 64, 76, 77, 79, 108] (see also [65, 87] and references therein for incremental subgradient methods for nondifferentiable convex optimization). The standard incremental gradient method updates x^k according to

$$x^{k+1} = x^k - \mu(k) \nabla f_{(k)_L}(x^k), \tag{2.5}$$

where $\mu(k)$ is a positive step size, possibly depending on k. Therefore, it is seen that the principal difference between the two methods is that the standard incremental gradient method uses only one of the components in order to generate an approximate

descent direction, whereas the IAG method uses the average of the L previously computed gradients. This property leads to convergence of the IAG method for fixed and sufficiently small positive step size μ. This is as contrasted to the standard incremental gradient method, whose convergence requires that the step size sequence $\mu(k)$ converge to zero.

Incremental gradient methods can be motivated by the observation that when the iterates are far from the eventual limit, the evaluation of a single gradient component is sufficient for generating an approximate descent direction. Hence, these methods lead to a significant reduction in the amount of required computations per iteration (see e.g. [16] section 1.5.2 and the discussion in [15]). The drawback of these methods, when using a constant step size, is that the iterates converge to a limit cycle and oscillate around a stationary point [76], unless restrictions of the type $\nabla f_l(x) = 0$, $l = 1, \ldots, L$ whenever $\nabla f(x) = 0$ are imposed [108]. Convergence for a diminishing step size has been established by a number of authors under different conditions [18,48,51, 64,76,77,79,108]. However, a diminishing step size usually leads to slow convergence near the eventual limit and requires exhaustive experimentation to determine how rapidly the step size must decrease in order to prevent scenarios in which the step size becomes too small when the iterates are far from the eventual limit (e.g. determining the constants a and b in step sizes of the form $\mu(k) = a/(k + b)$).

A hybrid between the steepest descent method and the incremental gradient method was studied in [15]. The hybrid method starts as an incremental gradient method and gradually becomes the steepest descent. This method requires a tuning parameter, which controls the transition between the two methods, to gradually increase with k to ensure convergence. When the tuning parameter increases sufficiently fast with the number of iterations, it is shown that the rate of convergence is linear. However, the question of determining the rate of transition between the

two methods still remains. For any fixed value of the tuning parameter, the hybrid method converges to a limit cycle, unless a diminishing step size is used, similar to the standard incremental gradient method.

The choice of the aggregated gradient d^k (2.3) for generating an approximate descent direction was mentioned in [51] in the context of adaptive step size methods, which require repeated evaluations of either the complete objective function $f(x)$ or its gradient. This requirement renders the methods proposed in [51] inapplicable to problems in sensor networks of interest to us or any other applications which require decentralized implementation, as will be explained in §3. In addition, as noted in [116], if $\nabla f_l(x)$, $l = 1, \ldots, L$, are not necessarily zero whenever $\nabla f(x) = 0$, the step size tends to zero, resulting in slow convergence.

The IAG method is closely related to Tseng's incremental gradient with momentum term [116], which is an incremental generalization of Polyak's heavy-ball method [91, p. 65] (also called the steepest descent with momentum term [17, p. 104]). Rewriting Tseng's method's update rule as

$$x^{k+1} = x^k - \mu(k) \sum_{l=0}^{k} \zeta^l \nabla f_{(k-l)_L}(x^{k-l}),$$

we see from (2.4) that the IAG method is a variation of this method with a truncated sum, $\zeta = 1$, and a constant step size. Similar to [51], the step size adaptation rule that leads to convergence in [116] requires repeated evaluations of the complete objective function $f(x)$ and its gradient. Hence, this method cannot be implemented in a distributed manner either. Furthermore, a linear convergence rate is established only under a certain growth property on the functions' gradients, which requires $\nabla f_l(x) = 0$, $l = 1, \ldots, L$, whenever $\nabla f(x) = 0$.

In contrast to the available methods, the IAG method has all four of the following

properties: (a) it evaluates a single gradient per iteration, (b) it uses a constant step size, (c) it is convergent (Proposition 2), and (d) it has global linear convergence rate for quadratic objective $f(x)$ (Proposition 3).

Finally, we note that the IAG method is reminiscent of other methods in various optimization problems, such as the incremental version of the Gauss-Newton method or the extended Kalman filter [10, 14, 36, 83], the distributed EM algorithm for maximum likelihood estimation [86, 90], the ordered subset and incremental optimization transfer for image reconstruction [2, 13, 28], and iterative methods for the convex feasibility problem [29, 30].

2.2 Convergence Analysis

In this section we present convergence proofs for two different function classes: (I) restricted Lipschitz and (II) quadratic. Under a Lipschitz condition and a bounded gradient assumption on $f_l(x)$, $l = 1, \ldots, L$ (Assumptions A.1 and A.2), we obtain an upper bound on the limit inferior of $||\nabla f(x^k)||$, which depends linearly on the step size μ. By imposing additional restrictions on the function $f(x)$ (Assumptions A.3 and A.4), we prove pointwise convergence of the method. There are many functions that satisfy Assumptions A.1–A.4. However, one important case does not satisfy these assumptions. This is the case when $f(x)$ and $f_l(x)$ are quadratic functions on $\mathbb{R}^p$. For this important case we provide a completely different convergence proof and show in addition that the convergence rate is globally linear.

For later reference, it will be useful to write (2.4) in a form known as the "gradient

method with errors" [18]:

$$\begin{aligned} x^{k+1} &= x^k - \mu\frac{1}{L}\left[\sum_{l=0}^{L-1}\nabla f_{(k-l)_L}(x^k) + \sum_{l=0}^{L-1}\nabla f_{(k-l)_L}(x^{k-l}) - \sum_{l=0}^{L-1}\nabla f_{(k-l)_L}(x^k)\right] \\ &= x^k - \mu\frac{1}{L}\left[\nabla f(x^k) + h^k\right], \end{aligned} \tag{2.6}$$

where

$$h^k = \sum_{l=1}^{L-1}\left[\nabla f_{(k-l)_L}(x^{k-l}) - \nabla f_{(k-l)_L}(x^k)\right]$$

is the error term in the calculation of the gradient at x^k. Also note that for all $k \geq 2L$ and $1 \leq l \leq L$,

$$x^{k-l} - x^k = \mu\frac{1}{L}\left(d^{k-1} + d^{k-2} + \ldots + d^{k-l}\right).$$

2.2.1 Case I

Assumptions A.1. *$\nabla f_l(x)$, $l = 1, \ldots, L$, satisfy a Lipschitz condition in $\mathbb{R}^p$, i.e. there is a positive number M_1 such that for all $x, \overline{x} \in \mathbb{R}^p$, $||\nabla f_l(x) - \nabla f_l(\overline{x})|| \leq M_1||x - \overline{x}||$, $l = 1, \ldots, L$.*

Assumption A.1 implies that $\nabla f(x)$ also satisfies a Lipschitz condition, that is, for all $x, \overline{x} \in \mathbb{R}^p$, $||\nabla f(x) - \nabla f(\overline{x})|| \leq M_2||x - \overline{x}||$, where $M_2 = LM_1$.

Assumptions A.2. *There exists a positive number M_3 such that for all $x \in \mathbb{R}^p$, $||\nabla f_l(x)|| \leq M_3$, $l = 1, \ldots, L$.*

Assumption A.2 implies that for all $x \in \mathbb{R}^p$, $||\nabla f(x)|| \leq M_4$, where $M_4 = LM_3$.

Lemma 1. *Let $\{s_k\}_{k\geq 1}$ be a sequence of non-negative real numbers satisfying for some fixed integer $L > 1$ and all $k \geq L$*

$$s_k \leq cQ(s_{k-1}, s_{k-2}, \ldots, s_{k-L+1}) + M,$$

where $0 < c < 1$, M is nonnegative, and $Q(s_{k-1}, s_{k-2}, \ldots, s_{k-L+1})$ is a linear form in the variables s_{k-1}, s_{k-2},...,s_{k-L+1}, whose coefficients are non-negative and the sum of the coefficients equals one. Then, $\limsup_{k\to\infty} s_k \leq \frac{M}{1-c}$.

Proof. Define the sequence $\{w_k\}_{k\geq 1}$ by $w_k = s_k$ for $1 \leq k \leq L-1$ and

$$w_k = cQ(w_{k-1}, w_{k-2}, \ldots, w_{k-L+1}) + M,$$

for $k \geq L$. Since $s_k \leq w_k$ for all k, if $\lim_{k\to\infty} w_k = \frac{M}{1-c}$ then

$$\limsup_{k\to\infty} s_k \leq \limsup_{k\to\infty} w_k = \lim_{k\to\infty} w_k = \frac{M}{1-c}.$$

To show that $\lim_{k\to\infty} w_k = \frac{M}{1-c}$, define the sequence $\{v_k\}_{k\geq 1}$ by $v_k = s_k - \frac{M}{1-c}$ for $1 \leq k \leq L-1$ and

$$v_k = cQ(v_{k-1}, v_{k-2}, \ldots, v_{k-L+1}),$$

for $k \geq L$. By this construction,

$$\begin{aligned} w_L &= cQ(\frac{M}{1-c} + v_{L-1}, \frac{M}{1-c} + v_{L-2}, \ldots, \frac{M}{1-c} + v_1) + M \\ &= c\frac{M}{1-c} + cQ(v_{L-1}, v_{L-2}, \ldots, v_1) + M = \frac{M}{1-c} + v_L, \end{aligned}$$

and, by induction, $w_k = \frac{M}{1-c} + v_k$ for all $k > L$. Therefore, if $\lim_{k\to\infty} v_k = 0$ then $\lim_{k\to\infty} w_k = \frac{M}{1-c}$. To show that $\lim_{k\to\infty} v_k = 0$, set $A = \max\{|v_1|, |v_2|, \ldots, |v_{L-1}|\}$. Hence,

$$|v_L| = c|Q(v_{L-1}, v_{L-2}, \ldots, v_1)| \leq cQ(|v_{L-1}|, |v_{L-2}|, \ldots, |v_1|) \leq cA.$$

Similarly, $|v_{L+1}| \leq cA$, and in general $|v_k| \leq cA$ for all $k \geq L$. Consider now v_{2L}.

Since $\max\{|v_{2L-1}|, |v_{2L-2}|, \ldots, |v_{L+1}|\} \leq cA$, we have

$$|v_{2L}| = c|Q(v_{2L-1}, v_{2L-2}, \ldots, v_{L+1})| \leq cQ(|v_{2L-1}|, |v_{2L-2}|, \ldots, |v_{L+1}|) \leq c^2 A,$$

and in general $|v_k| \leq c^2 A$ for all $k \geq 2L$. Similarly, we obtain $|v_k| \leq c^n L$ for all $k \geq nL$. Since $0 < c < 1$, we have $\lim_{n\to\infty} c^n = 0$, and therefore $\lim_{k\to\infty} v_k = 0$. □

Remark 1. *Lemma* 1 *can also be proven using concepts from dynamical systems. The sequence* w_k *is the output of an autoregressive linear system*

$$w_k = c\sum_{l=1}^{L-1} \alpha_k w_{k-l} + Mu(k-L),$$

where $u(k)$ *is the unit step function which equals one when* $k \geq 0$ *and zero otherwise, with initial condition* $w_k = s_k$ *for* $1 \leq k \leq L-1$*. Since the coefficients of the linear form are all positive and sum to one, and* $0 < c < 1$*, it is possible to show that the system is stable (bounded input bounded output) and the steady state response is* $\frac{M}{1-c}$ [92]*, i.e.,* $\lim_{k\to\infty} w_k = \frac{M}{1-c}$.

Lemma 2. *Under Assumption* A.1*, if* $||\nabla f(x^k)|| > \frac{||h^k||}{1-2\mu M_1}$*, and* $0 < 1 - 2\mu M_1 < 1$*, then* $f(x^k) > f(x^{k+1})$.

Proof. Assume that $||\nabla f(x^k)|| > \frac{||h^k||}{1-2\mu M_1}$. Then

$$\begin{aligned} ||d^k||^2 &= ||\nabla f(x^k) + h^k||^2 \leq 2||\nabla f(x^k)||^2 + 2||h^k||^2 \\ &< 2||\nabla f(x^k)||^2 + 2\frac{||h^k||^2}{1-2\mu M_1} < 4||\nabla f(x^k)||^2. \end{aligned}$$

By [16, Prop. A.24], if Assumption A.1 holds, then

$$f(x+y) - f(x) \leq y'\nabla f(x) + \frac{1}{2}M_2||y||^2.$$

Hence

$$\begin{aligned}
f(x^k) - f(x^{k+1}) &= f(x^k) - f(x^k - \mu\frac{1}{L}d^k) \\
&\geq \mu\frac{1}{L}d^{k\prime}\nabla f(x^k) - \frac{1}{2}M_2\mu^2\frac{1}{L^2}||d^k||^2 \\
&> \mu\frac{1}{L}(\nabla f(x^k) + h^k)'\nabla f(x^k) - \frac{1}{2}M_2\mu^2\frac{1}{L^2}4||\nabla f(x^k)||^2 \\
&= \mu\frac{1}{L}||\nabla f(x^k)||^2 + \mu\frac{1}{L}h^{k\prime}\nabla f(x^k) - 2M_2\mu^2\frac{1}{L^2}||\nabla f(x^k)||^2 \\
&\geq \mu\frac{1}{L}||\nabla f(x^k)||^2 - \mu\frac{1}{L}||h^k|| \cdot ||\nabla f(x^k)|| - 2M_2\mu^2\frac{1}{L^2}||\nabla f(x^k)||^2 \\
&= \frac{\mu}{L}||\nabla f(x^k)||(1 - 2\mu M_1)\left(||\nabla f(x^k)|| - \frac{||h^k||}{1 - 2\mu M_1}\right) \\
&> 0.
\end{aligned}$$

Lemma 3. *Set $\delta_0 = \mu M_2 M_3$. Under Assumptions* A.1 *and* A.2*, if $\mu M_2 < 1$, there exists K such that for all $k > K$, $||h^k|| < \delta_0$.*

Proof.

$$\begin{aligned}
||h^k|| &\leq \sum_{l=1}^{L-1} ||\nabla f_{(k-l)_L}(x^{k-l}) - \nabla f_{(k-l)_L}(x^k)|| \\
&\leq M_1 \sum_{l=1}^{L-1} ||x^{k-l} - x^k|| \\
&= \mu M_1 \frac{1}{L} \sum_{l=1}^{L-1} ||d^{k-1} + d^{k-2} + \ldots + d^{k-l}|| \\
&\leq \mu M_1 \frac{1}{L} \sum_{l=1}^{L-1} \left(||d^{k-1}|| + ||d^{k-2}|| + \ldots + ||d^{k-l}||\right) \\
&= \mu M_1 \frac{1}{L} \left[(L-1)||d^{k-1}|| + (L-2)||d^{k-2}|| + \ldots + ||d^{k-L+1}||\right] \\
&= \mu M_1 \frac{1}{L} \frac{L(L-1)}{2} \left[\frac{(L-1)||d^{k-1}|| + (L-2)||d^{k-2}|| + \ldots + ||d^{k-L+1}||}{L(L-1)/2}\right] \\
&= \mu M_1 \frac{L-1}{2} Q(||d^{k-1}||, ||d^{k-2}||, \ldots, ||d^{k-L+1}||),
\end{aligned}$$

where $Q(||d^{k-1}||, ||d^{k-2}||, \ldots, ||d^{k-L+1}||)$ is a linear form in the variables $||d^{k-1}||$, $||d^{k-2}||$, $\ldots$,$||d^{k-L+1}||$ whose coefficients, $\frac{L-1}{L(L-1)/2}$, $\frac{L-2}{L(L-1)/2}$, $\ldots$, $\frac{1}{L(L-1)/2}$, sum to one. Next we use $||d^k|| = ||\nabla f(x^k) + h^k|| \leq ||\nabla f(x^k)|| + ||h^k||$ to obtain

$$\begin{aligned}
||h^k|| &\leq \mu M_1 \frac{L-1}{2} Q(||h^{k-1}||, ||h^{k-2}||, \ldots, ||h^{k-L+1}||) \\
&\quad + \mu M_1 \frac{L-1}{2} Q(||\nabla f(x^{k-1})||, ||\nabla f(x^{k-2})||, \ldots, ||\nabla f(x^{k-L+1})||) \\
&\leq \mu M_1 \frac{L-1}{2} Q(||h^{k-1}||, ||h^{k-2}||, \ldots, ||h^{k-L+1}||) + \mu M_1 \frac{L-1}{2} M_3 \\
&< \mu \frac{M_2}{2} Q(||h^{k-1}||, ||h^{k-2}||, \ldots, ||h^{k-L+1}||) + \mu \frac{M_2}{2} M_3,
\end{aligned}$$

where Assumption A.2 was used in the second to last inequality. Hence, by Lemma 1, since $0 < \mu \frac{M_2}{2} < 1/2$, $\limsup_{k\to\infty} ||h^k|| \leq \frac{\mu \frac{M_2}{2} M_3}{1-\mu \frac{M_2}{2}}$. By using $\mu \frac{M_2}{2} < 1/2$, we obtain $\limsup_{k\to\infty} ||h^k|| < \mu M_2 M_3$ and the lemma follows. □

Proposition 1. *Under Assumptions* A.1 *and* A.2, *if* $f(x)$ *is bounded from below and* $\mu \max\{2M_1, M_2\} < 1$ *then,*

$$\liminf_{k\to\infty} ||\nabla f(x^k)|| \leq \frac{2M_2 M_3}{1 - 2\mu M_1} \mu.$$

Proof. Similar to the proof of Theorem 2.1 in [108]. □

Next, by imposing two additional assumptions, we prove that the IAG method converges with a constant step size to the minimum point of $f(x)$.

Assumptions A.3. *$f(x)$ has a unique global minimum at x^*. The Hessian $\nabla^2 f(x)$ is continuous and positive definite at x^*.*

Assumptions A.4. *For any sequence $\{t^k\}_{k=1}^{\infty}$ in $\mathbb{R}^p$, if $\lim_{k\to\infty} f(t^k) = f(x^*)$ or $\lim_{k\to\infty} ||\nabla f(t^k)|| = 0$, then $\lim_{k\to\infty} t^k = x^*$.*

There is an equivalent form of Assumption A.4: For each neighborhood $\mathcal{U}$ of x^* there exists $\eta > 0$ such that if $f(x) - f(x^*) < \eta$ or $||\nabla f(x)|| < \eta$, then $x \in \mathcal{U}$.

Remark 2. *Assumptions* A.3 *and* A.4 *are stronger than the assumptions usually made on* $f(x)$ *in the literature (see* [18] *for a summary of the available convergence proofs and the assumptions they require). However, our results hold for a constant step size and do not require that* $\nabla f_l(x) = 0$, $l = 1, \ldots, L$, *whenever* $\nabla f(x) = 0$. *In addition, note that there are non-convex functions that satisfy Assumption* A.4. *However, if* $f(x)$ *is strictly convex and takes a minimum in the interior of its domain* ($\mathbb{R}^p$), *then Assumption* A.4 *is automatically satisfied. In particular, if* $f(x)$ *satisfies Assumption* 3 *and is strictly convex, then Assumption* 4 *is satisfied. In fact, the implication* $\lim_{k\to\infty} f(t^k) = f(x^*) \Rightarrow \lim_{k\to\infty} t^k = x^*$ *is the statement of Corollary* 27.2.2 *from* [101]. *The implication* $\lim_{k\to\infty} ||\nabla f(t^k)|| = 0 \Rightarrow \lim_{k\to\infty} t^k = x^*$ *can be obtained as follows: Consider the function* $\nabla f : \mathbb{R}^p \to \mathbb{R}^p$. *The derivative* $(\nabla f)'$ *of this function is the Hessian* $\nabla^2 f$. *Since* $f(x)$ *is strictly convex,* $\det(\nabla f)' \neq 0$. *Therefore, by the Inverse Function Theorem, there are open neighborhoods* V *of* $x^* \in \mathbb{R}^p$ *and* W *of* $0 \in \mathbb{R}^p$ *such that* $\nabla f : V \to W$ *has a continuous inverse* $\gamma : W \to V$. *Let* $\{t^k\}_{k=1}^{\infty}$ *be a sequence such that* $\lim_{k\to\infty} ||\nabla f(t^k)|| = 0$. *Then there exists* k_0 *such that* $\nabla f(t^k) \in W$ *for all* $k \geq k_0$. *By Theorem B on page* 99 *in* [100], *since* $f(x)$ *is strictly convex,* ∇f *is one-to-one, i.e. if* $x \neq y$, *then* $\nabla f(x) \neq \nabla f(y)$. *It follows that* $t^k \in V$ *for all* $k \geq k_0$. *Now we have*

$$\begin{aligned} \lim_{k\to\infty} t^k &= \lim_{k\to\infty} \gamma\left(\nabla f(t^k)\right) \\ &= \gamma\left(\lim_{k\to\infty} \nabla f(t^k)\right) \\ &= \gamma(0) = x^*. \end{aligned}$$

Remark 3. *Unimodal functions which are convex in the neighborhood of their mini-*

mum and have bounded gradient are common in robust estimation [55]*. An example of a robust estimation objective function that satisfies Assumptions* 1-4 *is given in* §4.1*. Another important function which satisfies Assumptions* 1-4 *is the objective function minimized by the LogitBoost algorithm* [46] *(or adaptive logistic regression). To explain the components which are used to construct this objective function we include a short description (taken from* [32]*) of the supervised learning problem, and in particular, the problem of combining weak features. Let* $\{z_l, y_l\}_{l=1}^{L}$ *be a set of training examples, where each instance* z_l *takes values in an instance domain* $\mathcal{Z}$*, and each* y_l*, called the label, takes values in* $\{-1, +1\}$*. Given a set of* p *real-valued functions on* $\mathcal{Z}$*,* $h_1, h_2, \ldots, h_p$ *called features, the goal is to find a vector* $x \in \mathbb{R}^p$ *for which the sign of* $g_x(z_l) = \sum_{i=1}^{p} x_i h_i(z_l)$ *is a good predictor of* y_l*, for* $l = 1, \ldots, L$*. Let* M *be the* $L \times p$ *matrix whose* (l, i) *element is* $h_i(z_l)$*. The objective function* $f(x) : \mathbb{R}^p \to \mathbb{R}$ *minimized by the LogitBoost algorithm [32] is given by*

$$f(x) = \sum_{l=1}^{L} \log\left[1 + \exp\left(-y_l [Mx]_l\right)\right]. \tag{2.7}$$

where $[Mx]_l$ *is the* l*'th element of the vector* Mx*. It can be motivated as being a convex surrogate to the non-convex and non-differentiable* $0-1$ *loss function*

$$f(x) = \sum_{l=1}^{L} I(g_x(z_l) y_l \leq 0),$$

which is the number of labels that are not predicted correctly by the sign of $g_x(z_l)$*, or through the maximum likelihood method for estimating the conditional probability of* y_l *given* z_l*. It is shown below that in the non-separable case, i.e., when there exists no value of* x *for which* $\operatorname{sign}(g_x(z_l)) = y_l$*, for* $l = 1, \ldots, L$*, and when the features are linearly independent on the training set, i.e.,* $\operatorname{rank} M = p$*, the function* $f(x)$ (2.7)

satisfies Assumptions 1-4.

$$\frac{\partial}{\partial x_j}\log\left[1+\exp\left(-y_l[Mx]_l\right)\right]=\frac{\exp\left(-y_l[Mx]_l\right)}{1+\exp\left(-y_l[Mx]_l\right)}\left(-y_l h_j(z_l)\right)\leq|h_j(z_l)|.$$

Hence Assumption 2 *holds.*

$$\begin{aligned}\frac{\partial^2}{\partial x_j\partial x_k}\log\left[1+\exp\left(-y_l[Mx]_l\right)\right] &= \frac{\exp\left(-y_l[Mx]_l\right)}{\left[1+\exp\left(-y_l[Mx]_l\right)\right]^2}h_j(z_l)h_k(z_l)\\ &\leq |h_j(z_l)h_k(z_l)|.\end{aligned}$$

Hence Assumption 1 *holds. Let* $d_l(x)=\exp\left(-y_l[Mx]_l\right)/\left[1+\exp\left(-y_l[Mx]_l\right)\right]^2>0$. *Then,*

$$\frac{\partial^2 f(x)}{\partial x_j\partial x_k}=\sum_{l=1}^{L}d_l(x)M_{lj}M_{lk}.$$

To show that $\nabla f(x)$ *is positive definite for all* x, *consider* $\zeta^T\nabla f(x)\zeta$ *for some vector* $\zeta\in\mathbb{R}^p$*:*

$$\zeta^T\nabla f(x)\zeta=\sum_{j,k=1}^{p}\sum_{l=1}^{L}d_l(x)M_{lk}M_{lj}\zeta_k\zeta_j=\sum_{l=1}^{L}d_l(x)\left([M\zeta]_l\right)^2\geq 0$$

with equality if and only if $\zeta=0$, *by the assumption that* $\mathrm{rank}M=p$. *Hence the function* $f(x)$ *is strictly convex. Assume the training set* $\{z_l,y_l\}_{l=1}^{L}$ *is non-separable with respect to the features* $h_1,h_2,\ldots,h_p$, *i.e., for every* x *there exists at least one* l *for which* $y_l[Mx]_l<0$. *For any given* $x\neq 0$ *let* $I_1(x)=\{l:y_l[Mx]_l<0\}$, $I_2(x)=\{l:y_l[Mx]_l=0\}$, *and* $I_3(x)=\{l:y_l[Mx]_l>0\}$, *and note that* $I_1(x)$ *is nonempty by assumption. For a positive scalar* c, *we can write* $f(cx)$ *as the sum of*

three summations:

$$f(cx) = \sum_{l\in I_1(x)} \log\left\{1+\exp\left[-cy_l\sum_{i=1}^{p} x_i h_i(z_l)\right]\right\} + \sum_{l\in I_2(x)} \log 2 + \sum_{l\in I_3(x)} \log\left\{1+\exp\left[-cy_l\sum_{i=1}^{p} x_i h_i(z_l)\right]\right\}.$$

When $c\to\infty$,

$$\sum_{l\in I_1(x)} \log\left\{1+\exp\left[-cy_l\sum_{i=1}^{p} x_i h_i(z_l)\right]\right\} \to \infty$$

and

$$\sum_{l\in I_3(x)} \log\left\{1+\exp\left[-cy_l\sum_{i=1}^{p} x_i h_i(z_l)\right]\right\} \to 0.$$

Therefore, $\lim_{c\to\infty} f(cx)=\infty$*, for all* $x\neq 0$*. This implies that* $f(x)$ *has no directions of recession. A direction of recession is a non-zero vector* x^1 *such that* $f(x^2+cx^1)$ *is a non-increasing function of the scalar* c *for every choice of vector* x^2*. Hence by Theorem 27.1(d) in* [101, p.265] *the minimum set of* $f(x)$ *is non-empty. The minimum is unique by the strict convexity of* $f(x)$*. Therefore, Assumption* 3 *is also satisfied, and the strict convexity together with Assumption* 3 *imply Assumption* 4 *as well.*

The following lemma is well known.

Lemma 4. *Under Assumption* A.3*, there exists a neighborhood* $\mathcal{U}$ *of* x^* *and positive constants* A_1, A_2, B_1, B_2 *such that for all* $x\in\mathcal{U}$*,*

$$A_1||x-x^*||^2 \le f(x)-f(x^*) \le B_1||x-x^*||^2, \tag{2.8}$$

$$A_2||x-x^*||^2 \le ||\nabla f(x)||^2 \le B_2||x-x^*||^2. \tag{2.9}$$

Let $\mathcal{U}$ be a neighborhood of x^* for which inequalities (2.8) and (2.9) hold. By assumption A.4 there exists $\eta > 0$ such that $x \in \mathcal{U}$ if $f(x) - f(x^*) < \eta$ or $||\nabla f(x)|| < \eta$.

Lemma 5. *Set* $M_5 = \max\{3\sqrt{\frac{B_1 B_2}{A_1 A_2}}, \frac{2}{1-2\mu M_1}\}$ *and* $\lambda = \mu M_2 M_5$. *Under Assumptions* A.1, A.3, *and* A.4, *if there exist positive numbers* n_1 *and* δ *such that* $||h^k|| < \delta$ *for every* $k \geq n_1$, $3\delta < \eta$, $\frac{9B_1}{A_2}\delta^2 < \eta$, *and* $9\mu M_1 < 1$, *then*

1. *there exists a number* k_1 *such that* $||\nabla f(x^k)|| < M_5\delta$ *and* $||d^k|| < 2M_5\delta$ *for every* $k \geq k_1$, *and*

2. *there exists a number* n_2 *such that* $||h^k|| < \lambda\delta$ *for every* $k \geq n_2$.

Proof. First we show that there exists k such that $k \geq n_1$ and $||\nabla f(x^k)|| < \frac{2\delta}{1-2\mu M_1}$. In fact, if $||\nabla f(x^k)|| \geq \frac{2\delta}{1-2\mu M_1}$ for all $k \geq n_1$, then $||\nabla f(x^k)|| > \frac{2||h^k||}{1-2\mu M_1} \geq \frac{||h^k||}{1-2\mu M_1}$ for all $k \geq n_1$. By Lemma 2, the sequence $\{f(x^k)\}_{k=n_1}^{\infty}$is decreasing. Since it is bounded from below by $f(x^*)$, there exists $\lim_{k\to\infty} f(x^k)$. By replacing δ_0 with δ and $\max\{K_1, K_2\}$ with n_1 at the last argument of the proof of Proposition 1, we obtain a contradiction.

Let k_1 be the smallest natural number such that $k_1 \geq n_1$ and $||\nabla f(x^{k_1})|| \leq \frac{2\delta}{1-2\mu M_1}$. Without loss of generality, assume there exists k_2, the smallest natural number such that $k_2 > k_1$ and $||\nabla f(x^{k_2})|| > \frac{2\delta}{1-2\mu M_1}$. Let k_3 be the smallest natural number such that $k_3 > k_2$ and $||\nabla f(x^{k_3})|| \leq \frac{2\delta}{1-2\mu M_1}$. Let k_4 be the smallest natural number such that $k_4 > k_3$ and $||\nabla f(x^{k_4})|| > \frac{2\delta}{1-2\mu M_1}$. We define $k_5, k_6, \ldots$ in a similar manner.

For every natural m,

$$||d^{k_{2m}-1}|| \leq ||\nabla f(x^{k_{2m}-1})|| + ||h^{k_{2m}-1}|| \leq \frac{2\delta}{1-2\mu M_1} + \delta \leq \frac{3\delta}{1-2\mu M_1},$$

$$||x^{k_{2m}} - x^{k_{2m}-1}|| = \mu \frac{1}{L} ||d^{k_{2m}-1}|| \leq \frac{3\mu/L}{1-2\mu M_1}\delta,$$

and

$$\begin{aligned} ||\nabla f(x^{k_{2m}})|| &\leq ||\nabla f(x^{k_{2m}}) - \nabla f(x^{k_{2m}-1})|| + ||\nabla f(x^{k_{2m}-1})|| \\ &\leq M_2 ||x^{k_{2m}} - x^{k_{2m}-1}|| + \frac{2\delta}{1-2\mu M_1} \\ &\leq M_2 \frac{3\mu/L}{1-2\mu M_1}\delta + \frac{2}{1-2\mu M_1}\delta \\ &= \frac{2+3\mu M_1}{1-2\mu M_1}\delta < 3\delta, \end{aligned}$$

where we used $\mu < \frac{1}{9M_1}$ to obtain the last inequality.

Since $||\nabla f(x^{k_{2m}})|| < 3\delta < \eta$, $x^{k_{2m}} \in \mathcal{U}$ and we can use Lemma 4. We obtain

$$f(x^{k_{2m}}) - f(x^*) \leq B_1 ||x^{k_{2m}} - x^*|| \leq \frac{B_1}{A_2} ||\nabla f(x^{k_{2m}})||^2 < \frac{B_1}{A_2} 9\delta^2.$$

Let k be such that $k_{2m} \leq k < k_{2m+1}$. Then, by Lemma 2,

$$f(x^k) - f(x^*) < f(x^{k_{2m}}) - f(x^*) < 9\frac{B_1}{A_2}\delta^2.$$

Since $f(x^k) - f(x^*) < 9\frac{B_1}{A_2}\delta^2 < \eta$, $x^k \in \mathcal{U}$, and we can use Lemma 4. We obtain

$$||\nabla f(x^k)||^2 \leq B_2 ||x^k - x^*||^2 \leq \frac{B_2}{A_1}\left[f(x^k) - f(x^*)\right] < 9\frac{B_1 B_2}{A_1 A_2}\delta^2.$$

Thus, if k satisfies $k_{2m} \leq k < k_{2m+1}$, we have $||\nabla f(x^k)|| < 3\sqrt{\frac{B_1 B_2}{A_1 A_2}}\delta$. If k satisfies $k_{2m-1} \leq k < k_{2m}$, we have $||\nabla f(x^k)|| < \frac{2}{1-2\mu M_1}\delta$. Therefore for each $k \geq k_1$, $||\nabla f(x^k)|| < M_5\delta$ and therefore:

$$||d^k|| \leq ||\nabla f(x^k)|| + ||h^k|| \leq M_5\delta + \delta < 2M_5\delta.$$

Thus, if $k \geq k_1$, we have

$$\begin{aligned} ||\nabla f(x^k)|| &< M_5\delta \\ ||d^k|| &< 2M_5\delta. \end{aligned} \tag{2.10}$$

This proves the first part of the Lemma.

To prove the second part, we take $n_2 = k_1 + L - 1$. If $k \geq n_2$, then not only x^k but also $L-1$ previous terms of the sequence $\{x^k\}$ satisfy inequalities (2.10). Therefore, by following the steps in the proof of Proposition 1, we have for $k \geq n_2$

$$\begin{aligned} ||h^k|| &\leq \mu M_1 \frac{1}{L} \sum_{l=1}^{L-1} \left(||d^{k-1}|| + ||d^{k-2}|| + \ldots + ||d^{k-l}|| \right) \\ &< \mu M_1 \frac{1}{L} 2M_5\delta \sum_{l=1}^{L-1} \sum_{m=1}^{l} 1 = \mu M_1 \frac{1}{L} 2M_5\delta \frac{L(L-1)}{2} \\ &< \mu M_2 M_5 \delta = \lambda\delta. \end{aligned}$$

Thus $||h^k|| < \lambda\delta$. This proves the second part of Lemma 5. □

Remark 4. *A direct result of Lemma* 5 *is that under Assumptions* A.1-A.4, $||h^k|| \to 0$ *is a sufficient condition for the convergence of* x^k*, generated by any gradient method with errors* (2.6)*, to* x^*.

Proposition 2. *Under Assumptions* A.1, A.2, A.3, *and* A.4,
if $\mu < \min\{\frac{1}{9M_1}, \frac{1}{M_2M_5}, \frac{\eta}{3M_1M_3}, \frac{1}{3M_2M_3}\sqrt{\frac{A_2\eta}{B_1}}\}$, *then* $\lim_{k\to\infty} x^k = x^*$.

Proof. We prove Proposition 2 by repeated use of Lemma 5. We start with $\delta = \delta_0$. By applying Lemma 3, there exists K such that for all $k > K$, $||h^k|| < \delta_0$. After applying Lemma 5 r times we get a number n_r such that $||h^k|| < \delta_0\lambda^r$, $||\nabla f(x^k)|| < M_5\delta_0\lambda^r$, and $||d^k|| < 2M_5\delta_0\lambda^r$, for $k \geq n_r$. The inequality $\mu < \frac{1}{M_2M_5}$ is equivalent to

$0 < \lambda < 1$. Hence, $\lim_{k\to\infty} ||h^k|| = 0$, $\lim_{k\to\infty} ||d^k|| = 0$, and $\lim_{k\to\infty} ||\nabla f(x^k)|| = 0$, and by Assumption A.4, $\lim_{k\to\infty} x^k = x^*$.

Note that the inequality $\mu < \frac{1}{9M_1}$ was used in the proof of Lemma 5, and the inequalities $\mu < \frac{\eta}{3M_2M_3}$ and $\mu < \frac{1}{3M_2M_3}\sqrt{\frac{A_2\eta}{B_1}}$ are equivalent to $3\delta_0 < \eta$ and $\frac{9B_1}{A_2}\delta_0^2 < \eta$, respectively. □

2.2.2 Case II: Quadratic Case

In [76] it is shown that when applied to the objective function

$$f(x) = \frac{1}{2}(x - c_1)^2 + \frac{1}{2}(x - c_2)^2,$$

the standard incremental gradient method with a constant step size

$$x^{k+1} = x^k - \mu\nabla f_{(k)_L}(x^k)$$

converges to a limit cycle with limit points

$$x_1^*(\mu) = \frac{(1-\mu)c_1 + c_2}{2-\mu}, \qquad x_2^*(\mu) = \frac{(1-\mu)c_2 + c_1}{2-\mu},$$

whenever $0 < \mu < 1$. When implementing the IAG method one obtains

$$\begin{aligned} x^{k+1} &= x^k - \frac{\mu}{2}\left[\left(x^k - c_{(k)_2}\right) + \left(x^{k-1} - c_{(k-1)_2}\right)\right] \\ &= x^k - \frac{\mu}{2}\left[x^k + x^{k-1} - (c_1 + c_2)\right]. \end{aligned}$$

Subtracting $x^* = (c_1 + c_2)/2$, the unique minimum of $f(x)$, from both sides and denoting the error at the kth iteration by $e^k = x^k - x^*$, leads to the following error

form

$$e^{k+1} = e^k - \frac{\mu}{2}\left[e^k + e^{k-1}\right].$$

The characteristic polynomial of this linear system is $\lambda^2 - (1-\mu/2)\lambda + \mu/2$ and it is easy to show that the roots of this polynomial are inside the unit circle whenever $0 < \mu < 2$. Hence, when $0 < \mu < 2$, $e^k \to 0$, i.e., x^k converges to the unique minimum, in contrast to the standard incremental gradient method.

More generally, suppose that the functions f_l, $l = 1, \ldots, L$, have the following form

$$f_l(x) = \frac{1}{2}x'Q_l x - c_l'x, \quad l = 1, \ldots, L, \tag{2.11}$$

where Q_l are given symmetric matrices, c_l are given vectors, and $\sum_{l=1}^L Q_l$ is positive definite. Under this assumption, the function $f(x) = \sum_{l=1}^L f_l(x)$ is strictly convex, has its minimum point at

$$x^* = (\sum_{l=1}^L Q_l)^{-1} \sum_{l=1}^L c_l, \tag{2.12}$$

and x^* is the only stationary point of $f(x)$.

Proposition 3. *For sufficiently small μ, $\lim_{k\to\infty} x^k = x^*$ and the rate of convergence of the IAG method* (1.4) *is linear.*

Proof. Plugging (2.11) in (2.4), the IAG method becomes

$$x^{k+1} = x^k - \mu\left[\sum_{l=0}^{L-1} Q_{(k-l)_L} x^{k-l} - c_{(k-l)_L}\right] = x^k - \mu\sum_{l=0}^{L-1} Q_{(k-l)_L} x^{k-l} + \mu c,$$

where $c = \sum_{l=1}^L c_l$, and the factor $\frac{1}{L}$ was absorbed into μ to simplify the notation. Subtracting x^* (2.12) from both sides and adding and subtracting x^* inside the

parentheses, we obtain

$$x^{k+1} - x^* = x^k - x^* - \mu \sum_{l=0}^{L-1} Q_{(k-l)_L}(x^{k-l} - x^* + x^*) + \mu c.$$

Denoting the error at the kth iteration by $e^k = x^k - x^*$ and the substitution of (2.12) for x^* lead to the following error form

$$e^{k+1} = e^k - \mu \sum_{l=0}^{L-1} Q_{(k-l)_L} e^{k-l}.$$

This relation between a new error and the previous errors can be seen as a periodically time varying linear system. To analyze its stability, which will lead to the convergence result, it is useful to consider L iterations as one iteration [82]. This can be seen as down-sampling the original system by a factor of L, which leads to a time invariant system of a lower sampling rate. Without loss of generality, consider the case where $k = NL$ for some integer N, i.e. $k+1$ corresponds to the first iteration of a new cycle. In this case we have

$$\begin{aligned} e^{k+1} &= e^k - \mu \sum_{l=0}^{L-1} Q_{(k-l)_L} e^{k-l} = e^k - \mu \begin{bmatrix} Q_L & Q_{L-1} & Q_{L-2} & \dots & Q_1 \end{bmatrix} \overline{e}^k \\ &= \begin{bmatrix} I_p - \mu Q_L & -\mu Q_{L-1} & -\mu Q_{L-2} & \dots & -\mu Q_1 \end{bmatrix} \overline{e}^k, \end{aligned}$$

where I_p is the $p \times p$ identity matrix and

$$\overline{e}^k = \begin{bmatrix} e^k \\ e^{k-1} \\ \vdots \\ e^{k-L+1} \end{bmatrix}.$$

Similarly,

$$\begin{aligned} e^{k+2} &= e^{k+1} - \mu \sum_{l=0}^{L-1} Q_{(k+1-l)L} e^{k+1-l} \\ &= e^{k+1} - \mu \left[\begin{array}{ccccc} Q_1 & Q_L & Q_{L-1} & \dots & Q_2 \end{array} \right] \overline{e}^{k+1} \\ &= \left[\begin{array}{ccccc} I_p - \mu Q_1 & -\mu Q_L & -\mu Q_{L-1} & \dots & -\mu Q_2 \end{array} \right] \overline{e}^{k+1}, \end{aligned}$$

and finally

$$\begin{aligned} e^{k+L} &= e^{k+L-1} - \mu \sum_{l=0}^{L-1} Q_{(k+L-1-l)L} e^{k+L-1-l} \\ &= e^{k+L-1} - \mu \left[\begin{array}{ccccc} Q_{L-1} & Q_{L-2} & Q_{L-3} & \dots & Q_L \end{array} \right] \overline{e}^{k+L-1} \\ &= \left[\begin{array}{ccccc} I_p - \mu Q_{L-1} & -\mu Q_{L-2} & -\mu Q_{L-3} & \dots & -\mu Q_L \end{array} \right] \overline{e}^{k+L-1}. \end{aligned}$$

This leads to the relation

$$\overline{e}^{k+L} = M_L \overline{e}^{k+L-1},$$

where

$$M_L = \left[\begin{array}{ccccc} I_p - \mu Q_{L-1} & -\mu Q_{L-2} & \dots & -\mu Q_1 & -\mu Q_L \\ I_p & 0_p & \dots & 0_p & 0_p \\ 0_p & I_p & \dots & 0_p & 0_p \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0_p & 0_p & \dots & I_p & 0_p \end{array} \right],$$

where 0_p denotes the $p \times p$ zero matrix. Taking another step we have

$$\overline{e}^{k+L} = M_L M_{L-1} \overline{e}^{k+L-2},$$

where

$$M_{L-1} = \begin{bmatrix} I_p - \mu Q_{L-2} & -\mu Q_{L-3} & \dots & -\mu Q_L & -\mu Q_{L-1} \\ I_p & 0_p & \dots & 0_p & 0_p \\ 0_p & I_p & \dots & 0_p & 0_p \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0_p & 0_p & \dots & I_p & 0_p \end{bmatrix},$$

and finally, by induction,

$$\overline{e}^{k+L} = M_L M_{L-1} \dots M_1 \overline{e}^k,$$

where

$$M_1 = \begin{bmatrix} I_p - \mu Q_L & -\mu Q_{L-1} & \dots & -\mu Q_2 & -\mu Q_1 \\ I_p & 0_p & \dots & 0_p & 0_p \\ 0_p & I_p & \dots & 0_p & 0_p \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0_p & 0_p & \dots & I_p & 0_p \end{bmatrix}.$$

Denoting $M = M_L M_{L-1} \dots M_1$, we have $\overline{e}^{k+L} = M\overline{e}^k$, and in general $\overline{e}^{k+nL} = M^n \overline{e}^k$. Therefore, if for sufficiently small $\mu > 0$ the eigenvalues of M are inside the unit circle, then $\lim_{n\to\infty} \overline{e}^{k+nL} = 0_{pL\times 1}$, where $0_{pL\times 1}$ is a $pL \times 1$ zero vector, i.e. the method converges to the minimum of the function $f(x)$ and the convergence rate is linear.

To prove that the eigenvalues of M are inside the unit circle, set

$$A = \begin{bmatrix} I_p & 0_p & \dots & 0_p & 0_p \\ I_p & 0_p & \dots & 0_p & 0_p \\ 0_p & I_p & \dots & 0_p & 0_p \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0_p & 0_p & \dots & I_p & 0_p \end{bmatrix},$$

and

$$B_k = \begin{bmatrix} Q_{(k-1)_L} & Q_{(k-2)_L} & \dots & Q_{(k+1)_L} & Q_k \\ 0_p & 0_p & \dots & 0_p & 0_p \\ 0_p & 0_p & \dots & 0_p & 0_p \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0_p & 0_p & \dots & 0_p & 0_p \end{bmatrix}, \quad k = 1, \dots, L,$$

so that $M_k = A - \mu B_k$ and $M = (A - \mu B_L)(A - \mu B_{L-1}) \dots (A - \mu B_1)$. Hence,

$$M = A^L - \mu \quad \big(B_L A^{L-1} + A B_{L-1} A^{L-2} + A^2 B_{L-2} A^{L-3} + \dots + A^{L-2} B_2 A + A^{L-1} B_1\big) + \mu^2 C(\mu),$$

where $C(\mu)$ is a $Lp \times Lp$ matrix whose elements are polynomials in μ.

Note that pre-multiplying a matrix by A will duplicate the first row of $p \times p$ matrices and will shift the rest of the rows down, discarding the last p rows. Post-multiplying by A will add the second column of $p \times p$ matrices to the first one and will shift the rest of the columns to the left, inserting a block of $p \times p$ zero matrices

to the last column. It follows that

$$A^L = \begin{bmatrix} I_p & 0_p & \dots & 0_p & 0_p \\ I_p & 0_p & \dots & 0_p & 0_p \\ I_p & 0_p & \dots & 0_p & 0_p \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ I_p & 0_p & \dots & 0_p & 0_p \end{bmatrix},$$

and

$$A^{L-k}B^kA^{k-1} = \begin{bmatrix} W_1(k) & 0_{(L-k+1)p\times(k-1)p} \\ 0_{(k-1)p\times(L-k+1)p} & 0_{(k-1)p\times(k-1)p} \end{bmatrix},$$

where $W_1(k)$ is a $(L-k+1)p \times (L-k+1)p$ matrix whose elements are

$$W_1(k) = \begin{bmatrix} \sum_{l=0}^{k-1} Q_{(l)_L} & Q_{L-1} & \dots & Q_k \\ \vdots & \vdots & & \vdots \\ \sum_{l=0}^{k-1} Q_{(l)_L} & Q_{L-1} & \dots & Q_k \end{bmatrix}.$$

Therefore, the characteristic polynomial $F(\mu, \lambda)$ of M is

$$F(\mu,\lambda) = \det\left(M - \lambda I_{Lp}\right) = \det\left(A^L - \mu\sum_{k=1}^{L} A^{L-k}B^kA^{k-1} - \lambda I_{Lp} + \mu^2 C(\mu)\right).$$

The first p columns of $\left(A^L - \mu \sum_{k=1}^{L} A^{L-k} B^k A^{k-1} - \lambda I_{Lp} + \mu^2 C(\mu)\right)$ are

$$\begin{bmatrix} (1-\lambda)I_p - \mu\left[LQ_L + (L-1)Q_1 + \ldots + Q_{L-1}\right] + \mu^2 C_{11} \\ I_p - \mu\left[(L-1)Q_L + (L-2)Q_1 + \ldots + Q_{L-2}\right] + \mu^2 C_{21} \\ I_p - \mu\left[(L-2)Q_L + (L-3)Q_1 + \ldots + Q_{L-3}\right] + \mu^2 C_{31} \\ \vdots \\ I_p - \mu\left(2Q_L + Q_1\right) + \mu^2 C_{L-1\,1} \\ I_p - \mu Q_L + \mu^2 C_{L1} \end{bmatrix},$$

the second p columns are

$$\begin{bmatrix} -(L-1)\mu Q_{L-1} + \mu^2 C_{12} \\ -(L-1)\mu Q_{L-1} - \lambda I_p + \mu^2 C_{22} \\ -(L-2)\mu Q_{L-1} + \mu^2 C_{32} \\ \vdots \\ -2\mu Q_{L-1} + \mu^2 C_{L-1\,2} \\ -\mu Q_{L-1} + \mu^2 C_{L2} \end{bmatrix},$$

the next $(L-3)p$ columns are

$$\begin{bmatrix} -(L-2)\mu Q_{L-2} + \mu^2 C_{13} & \ldots & -2\mu Q_2 + \mu^2 C_{1\,L-1} \\ -(L-2)\mu Q_{L-2} + \mu^2 C_{23} & \ldots & -2\mu Q_2 + \mu^2 C_{2\,L-1} \\ -(L-2)\mu Q_{L-2} - \lambda I_p + \mu^2 C_{33} & \ldots & -2\mu Q_2 + \mu^2 C_{3\,L-1} \\ \vdots & & \vdots \\ -2\mu Q_{L-2} + \mu^2 C_{L-1\,3} & \ldots & -2\mu Q_2 - \lambda I_p + \mu^2 C_{L-1\,L-1} \\ -\mu Q_{L-2} + \mu^2 C_{L3} & \ldots & -\mu Q_2 + \mu^2 C_{L\,L-1} \end{bmatrix},$$

and the last p columns are

$$\begin{bmatrix} -\mu Q_1 + \mu^2 C_{1L} \\ -\mu Q_1 + \mu^2 C_{2L} \\ -\mu Q_1 + \mu^2 C_{3L} \\ \vdots \\ -\mu Q_1 + \mu^2 C_{L-1\,L} \\ -\mu Q_1 - \lambda I_p + \mu^2 C_{LL} \end{bmatrix},$$

where C_{ij}, $i, j = 1, \ldots L$ are $p \times p$ matrices whose entrees are polynomials in μ.

It is easy to see that if $\mu = 0$, then $F(0, \lambda) = (-1)^{Lp}\lambda^{Lp-p}(\lambda - 1)^p$. Hence, if $\mu = 0$, we have an eigenvalue 0 of multiplicity $Lp - p$ and an eigenvalue 1 of multiplicity p. If μ is close enough to zero, the 0-eigenvalues will be close to the origin and therefore inside the unit circle. We need to prove that for sufficiently small positive μ, all the 1-eigenvalues will be inside the unit circle. Let $\lambda = \lambda(\mu)$ be a smooth function expressing the dependence of one of the 1-eigenvalues on μ. We will prove that $\frac{\mathrm{d}\lambda}{\mathrm{d}\mu}(0^+) < 0$. It will be enough for our purposes since it will show that the trajectory $\lambda = \lambda(\mu)$ is entering the unit circle, and hence $\lambda(\mu)$ is inside the unit circle for sufficiently small positive μ.

By the definition of $\lambda(\mu)$, $\lambda(0+) = 1$ and $F(\mu, \lambda(\mu)) = 0$ for all μ. It follows that

$$\frac{\mathrm{d}^p F(\mu, \lambda(\mu))}{\mathrm{d}\mu^p} = 0. \tag{2.13}$$

To calculate the left side of (2.13), we use the formula for the derivative of a determinant [66]. Note that substituting $\mu = 0$ and $\lambda = 1$ into each of the first p rows of the matrix $M - \lambda I_{Lp}$ leads to a row in which all of the entrees are zeros and therefore the determinant has a zero value. Therefore the only non-zero terms in $\frac{\mathrm{d}^p F(\mu, \lambda(\mu))}{\mathrm{d}\mu^p}$

after substituting $\mu = 0$ and $\lambda = 1$ (more precisely, taking $\mu \to 0^+$) are the terms with the first derivatives in the first p rows (there are $p!$ such terms). Hence taking the pth derivative is reduced to taking the first derivative of each of the first p rows. Substituting $\lambda = 1$ and $\mu \to 0^+$ we obtain

$$\frac{\mathrm{d}^p F(\mu, \lambda(\mu))}{\mathrm{d}\mu^p} = p! \det \begin{bmatrix} W_2 & W_3 \\ W_4 & -I_{(L-1)p\times(L-1)p} \end{bmatrix} = 0,$$

where $W_2 = -\lambda'(0^+)I_p - \sum_{k=0}^{L-1}(L-k)Q_{(k)_L}$,

$$W_3 = \begin{bmatrix} -(L-1)Q_{L-1} & -(L-2)Q_{L-2} & \dots & -2Q_2 & -Q_1 \end{bmatrix},$$

and $W_4 = [I_p\, I_p\, \dots I_p]^T$. Add all columns of $p \times p$ matrices to the first column of $p \times p$ matrices to obtain

$$\det \begin{bmatrix} W_5 & W_3 \\ 0_{(L-1)p\times p} & -I_{(L-1)p\times(L-1)p} \end{bmatrix} = 0,$$

where $W_5 = -\lambda'(0^+)I_p - L\sum_{k=1}^{L} Q_k$. Calculating the last determinant gives

$$\det \left[L\sum_{k=1}^{L} Q_k + \lambda'(0^+)I_p \right] = 0.$$

The last equation shows that $-\lambda'(0^+)$ is an eigenvalue of the matrix $L\sum_{k=1}^{L} Q_k$. Since $L\sum_{k=1}^{L} Q_k$ is positive definite, $-\lambda'(0^+) > 0$ and therefore $\lambda'(0^+) < 0$. This proves that for sufficiently small $\mu > 0$ the eigenvalues of the matrix M are strictly inside the unit circle and hence the sequence x^k converges to x^* and the convergence rate is linear. □

2.3 Initialization and Distributed Implementation

As mentioned in §1, the IAG method is initiated with L points, $x^1, x^2, \ldots, x^L$. Possible initialization strategies include setting $x^1 = x^2 = \ldots = x^L$ or generating the initial points using a single cycle of the standard incremental gradient method (2.5). Another possibility is the following. Given x^1, compute $d^1 = \nabla f_1(x^1)$. Then, for $1 \leq k \leq L-1$,

$$\begin{aligned} x^{k+1} &= x^k - \mu \frac{1}{k} d^k, \\ d^{k+1} &= d^k + \nabla f_{(k+1)_L}(x^{k+1}). \end{aligned} \tag{2.14}$$

Therefore, after $L-1$ iterations we obtain $x^1, \ldots, x^L$ and $d^L = \sum_{l=1}^{L} \nabla f_l(x^l)$.

The key feature of the IAG method that makes it suitable for wireless sensor networks applications is that it can be implemented in a distributed manner. Consider a distributed system of L processors enumerated over $1, 2, \ldots, L$, each of which has access to one of the functions $f_l(x)$. The initialization (2.14) begins with x^1 at processor 1. Then, processor 1 sets $d^1 = \nabla f_1(x^1)$ and transmits x^1 and d^1 to processor 2. Upon receiving x^{k-1} and d^{k-1} from processor $k-1$, processor k calculates x^k and d^k according to (2.14) and transmits them to processor $k+1$. The initialization phase is completed when processor L, upon receiving x^{L-1} and d^{L-1} from processor $L-1$, computes x^L and d^L according to (2.14) and transmits them to processor 1.

Once the initialization phase is completed, the algorithm progresses in a cyclic manner. Upon receiving x^{k-1} and d^{k-1} from processor $(k-1)_L$, processor $(k)_L$ computes x^k and d^k according to (2.2) and (2.3), respectively, and transmits them to processor $(k+1)_L$. Note that $\nabla f_{(k)_L}(x^{k-L})$ in (2.3) is available at processor $(k)_L$, since it was the last gradient computed at that processor. Therefore, the only gradient

computation at processor $(k)_L$ is $\nabla f_{(k)_L}(x^k)$. At no phase of the algorithm do the processors share information regarding the complete function $f(x)$ or its gradient $\nabla f(x)$.

2.4 Application to Wireless Sensor Networks

There are two motivations to use the IAG method: (a) reduced computational burden due to the evaluation of a single gradient per iteration compared to L gradients required for the steepest descent method; and (b) the possibility of a distributed implementation of the method in which each component has access to one of the functions $f_l(x)$. The second item has been shown to be very useful in the context of wireless sensor networks [96]. Wireless sensor networks provide means for efficient large scale monitoring of large areas [113]. Often the ultimate goal is to estimate certain parameters based on measurements that the sensors collect, giving rise to an optimization problem. If measurements from distinct sensors are modelled as statistically independent, the estimation problem takes the form of (2.1), where $f_l(x)$ is indexed by the measurements available at sensor l (see e.g. [24, 90, 94, 95] and references therein). When transmitting the complete set of data to a central processor is impractical due to bandwidth and power constraints, the IAG method can be implemented in a distributed manner as described in §3. In the following sections we consider two such estimation problems.

2.4.1 Robust Estimation

One of the benefits of a wireless sensor network is the ability to deploy a large number of low cost sensors to densely monitor a certain area [113]. Because low cost sensors have limited reliability, the system must be designed to be robust to the

possibility of individual sensor failures. In estimation tasks, this means that some of the sensors will contribute unreliable measurements, namely outliers. In [94] the authors suggest the use of robust statistics to alleviate the influence of outliers in the data (see [55] or, specifically in the context of optimization, see [91, p. 347]). The robust statistics framework uses objective functions that give less weight to outliers. A common objective function used to this end is the function "Fair" [99, p. 110], given by

$$g(x) = c^2 \left[\frac{|x|}{c} - \log\left(1 + \frac{|x|}{c}\right) \right]. \tag{2.15}$$

We use the function "Fair" rather than the more common Huber [55] since the later is not strictly convex.

Following [94] we simulate a sensor network for measuring pollution levels and assume that a certain percentage of the sensors are damaged and provide unreliable measurements. Each sensor collects a single noisy measurement of the pollution level and the estimate of the average pollution level is found by minimizing the objective function defined by

$$f(x) = \sum_{l=1}^{L} f_l(x), \tag{2.16}$$

where $x \in \mathbb{R}$, and

$$f_l(x) = \frac{1}{L} g(x - y_l),$$

where y_l is the measurement collected by sensor l. There were $L = 50$ sensors in the simulation. To reflect the possibility of faulty sensors, half of the samples were generated according to a Gaussian distribution with mean $m_1 = 10$ and unit variance ($\sigma_1^2 = 1$) and the other half were generated according to a Gaussian distribution with mean $m_2 = 10$ and ten times higher variance ($\sigma_2^2 = 10$). The coefficient c in (2.15) was chosen to be 10.

For positive x, the first derivative of $g(x)$ is $\frac{x}{1+x/c}$, and for negative x it is $\frac{x}{1-x/c}$.

Hence, $g'(0+) = g'(0-) = 0$. The continuity of $g(x)$ implies then that it is differentiable at zero despite the term $|x|$. Therefore, the first derivative of $g(x)$ is $\frac{x}{1+|x|/c}$, it is continuous, and it is bounded by c. Considering positive and negative x's separately, also shows that $g''(0+) = g''(0-) = 1$, and that in general, the second derivative of $g(x)$ is $\frac{1}{(1+|x|/c)^2}$ which is bounded by 1. Hence both Assumptions A.1 and A.2 hold. In addition, since $\frac{1}{(1+|x|/c)^2}$ is strictly positive, $g(x)$ is strictly convex, and therefore $f(x)$ is strictly convex as well. Since both $\lim_{x\to\infty} f(x)$ and $\lim_{x\to-\infty} f(x)$ diverge to ∞, $f(x)$ has no directions of recession, and therefore, by Theorem 27.1(d) in [101, p.265], the minimum set of $f(x)$ is non-empty. The minimum is unique by the strict convexity of $f(x)$. Since $g''(x)$ is continuous and positive everywhere Assumption A.3 is satisfied. The strict convexity of $f(x)$ implies that Assumption A.4 holds as well (see Remark 2).

Both the standard incremental gradient method (2.5) with a constant step size $\mu(k) = \mu$ (abbreviated as "IG" in the figures) and the IAG method with the initialization (2.14) were implemented with several choices of step size μ. The initial point x^1 was set to 0. In Fig. 2.1 the trajectories of the two methods are presented. The solid straight line corresponds to the minimum point x^*. It is seen that when the step size is sufficiently small, IAG increases more rapidly towards x^* than the standard incremental gradient in the early iterations. Furthermore, as predicted by the theory, IAG converges to the true limit, whereas incremental gradient method converges to a limit cycle. For a larger step size the IAG method overshots due to its heavy ball characteristic (2.4). When the step size is too large, the IAG method no longer converges but the incremental gradient method still converges to a limit cycle. We have observed this behavior for other values of the parameters m_1, m_2, σ_1^2, σ_2^2, c as well.

We also compared the IAG method with the incremental gradient method with

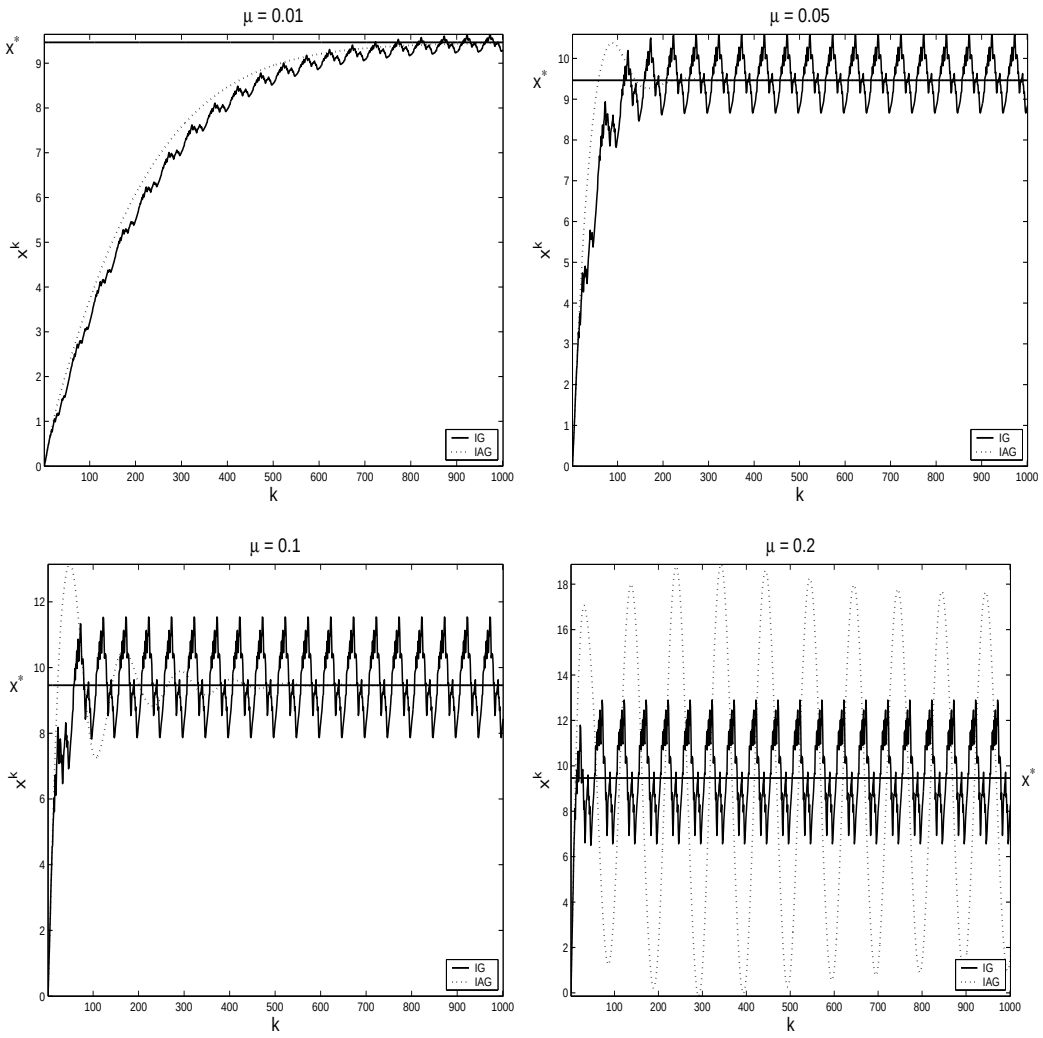

Figure 2.1: Trajectories taken by the IG and IAG methods for the robust "Fair" estimation problem.

a diminishing step size, with Bertsekas' hybrid method [15], and with Tseng's incremental gradient with momentum [116] in terms of number of iterations to convergence. To optimize the performance of the incremental gradient method with a diminishing step size, a relatively large constant step size $\mu = 0.2$ is used until convergence to a limit cycle is detected, and then the diminishing step size is $\mu(k) = .2\mu/(\widetilde{k} - k)$, where $\widetilde{k}$ is the first iteration in which a limit cycle is detected. Convergence to a limit cycle is declared when $|x^k - x^{k-L}| < .01$ for k a multiple of L. To describe the parameters used in the hybrid method, we switch to the notation in [15]. We set $\gamma = 0.05$ and $\alpha(\mu)$ as defined in Eq. (47) in [15], with $\phi(\mu) = \zeta(1-\mu)$, where $\zeta = 2.5$. The transition parameter μ is kept at zero, i.e., the iterates are identical to the incremental gradient method until convergence to a limit cycle is detected as described above. Once a limit cycle is detected, μ is updated after every cycle according to $\mu := 1.5\mu + 0.3$, i.e., $\widehat{n} = 1$. These parameters seemed to optimize the performance of the hybrid method. The parameters of the incremental gradient with momentum term where set according to the recommendation in [116], which seemed to optimize the performance of the method in our application as well. In particular, we set $\epsilon_0 = 1$, $\epsilon_1 = \epsilon_2 = 0.00001$, $\epsilon_3 = 1000$, $\eta = 1.5f(x_1^0) + 100$, $\rho = \infty$, $\omega = 0.5$, $\zeta = 0.8$, and $\lambda_1 + \lambda_2 + \ldots + \lambda_m = 1$. For the IAG method we set $\mu = 0.05$. The convergence point was specified to be the first iteration for which all subsequent iterations satisfy $|x^k - x^*| < \epsilon$. Since the IAG and the hybrid methods outperform the incremental gradient method with a diminishing step size and the incremental gradient with momentum term by a large margin, ϵ was specified to be 0.01 for the IAG and the hybrid method and 0.1 for the incremental gradient method with a diminishing step size and the incremental gradient with momentum term. The average number of iterations until convergence and its standard deviation were estimated from 100 Monte Carlo simulations and are summarized in Table 2.1. The trajectory taken by

	IAG $\epsilon = 0.01$	Hybrid $\epsilon = 0.01$	IG Diminishing Stepsize $\epsilon = 0.1$	IG Momentum Term $\epsilon = 0.1$
mean	290	589	601	2063
std	23	135	258	919

Table 2.1: Number of iterations to convergence.

the different methods in one of these simulations is presented in Fig. 2.2. It is seen that for this application, the IAG method performs best. Further experimentation is required to make more general conclusions.

2.4.2 Source Localization

This section presents a simulation of a sensor network for localizing a source that emits acoustic waves. L sensors are distributed on the perimeter of a field at known spatial locations, denoted r_l, $l = 1, \ldots, L$, where $r_l \in \mathbb{R}^2$. Each sensor collects a noisy measurement of the acoustic signal transmitted by the source, denoted y_l, at an unknown location x. Based on a far-field assumption and an isotropic acoustic wave propagation model [31, 75, 94, 104, 105], the problem of estimation of source location can be formulated as a non-linear least squares problem. The objective function is again of the form (2.16), but now

$$f_l(x) = \left(y_l - g(||r_l - x||^2)\right)^2, \tag{2.17}$$

$x \in \mathbb{R}^2$, and

$$g(z) = \begin{cases} A/z & : \quad z \geq A/\epsilon \\ 2\epsilon - \epsilon^2 z/A & : \quad z < A/\epsilon \end{cases}. \tag{2.18}$$

In (2.17) $g(\cdot)$ models the received signal strength as a function of the squared distance. In (2.18) A is a known constant characterizing the source's signal strength. For $z \geq A/\epsilon$ (far-field source), the source's signal strength has isotropic attenuation as

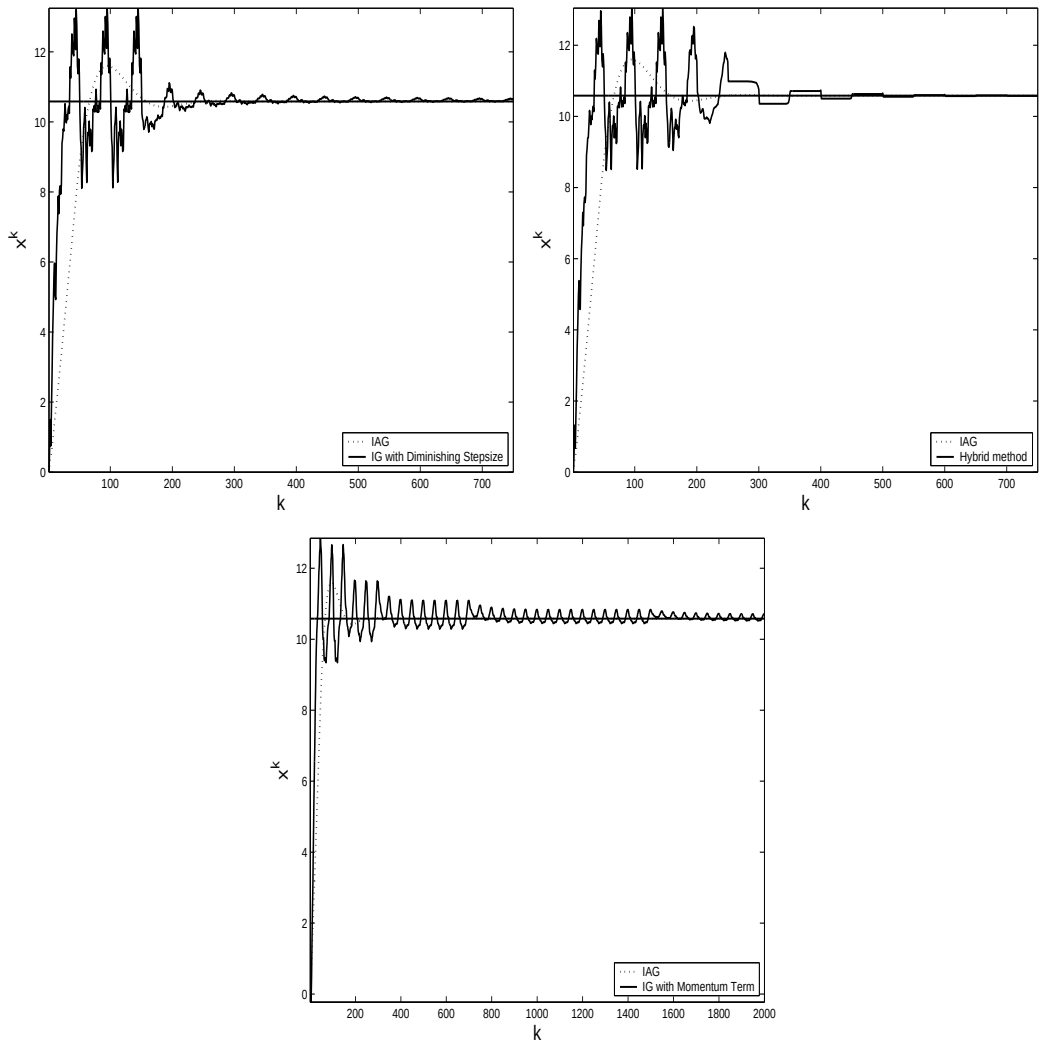

Figure 2.2: IAG compared to IG with diminishing step size, to the hybrid method, and to IG with momentum term.

an inverse function of the squared distance, while for $z < A/\epsilon$ (near-field source), the attenuation is linear in the squared distance. It is easy to see that Assumptions A.1 and A.2 are satisfied and therefore, Proposition 1 holds. Clearly, since $f(x)$ is multimodal in this case, Assumptions A.3 and A.4 cannot hold. However, it was observed in our experiments that when the source is sufficiently distant from the sensors, the objective function has a single minimum inside the observed field (See Fig. 2.4 for a contour plot of the objective function) and, when initiated not too far from the minimum point, the IAG method has good convergence properties. This suggests the possible application of the IAG method under weaker assumptions than those considered in this paper, and motivates further investigation into its properties.

In the numerical experiment, $L = 32$ sensors are distributed equidistantly on the perimeter of a 100×100 field. The source is located at the point $[60, 60]$ and emits a signal with strength $A = 1000$. The sensors' noisy measurements were generated according to a Gaussian distribution with a mean equal to the true signal power and unit variance. Both the incremental gradient method with a constant step size and the IAG method with the initialization (2.14) were initiated at the point $[40, 40]$. The error term $||x^k - x^*||$ as a function of the iteration number is presented in Fig. 2.3 for two choices of step size. The actual path taken by the methods for step size $\mu = 10$ is presented in Fig. 2.4, where the asterisk denotes the true minimum point of the objective function. It is seen that, as the theory predicts, the incremental gradient method exhibits oscillations near the eventual limit, whereas the IAG method converges to the minimum. In this scenario, the IAG method outperforms the IG method at early iterations as well.

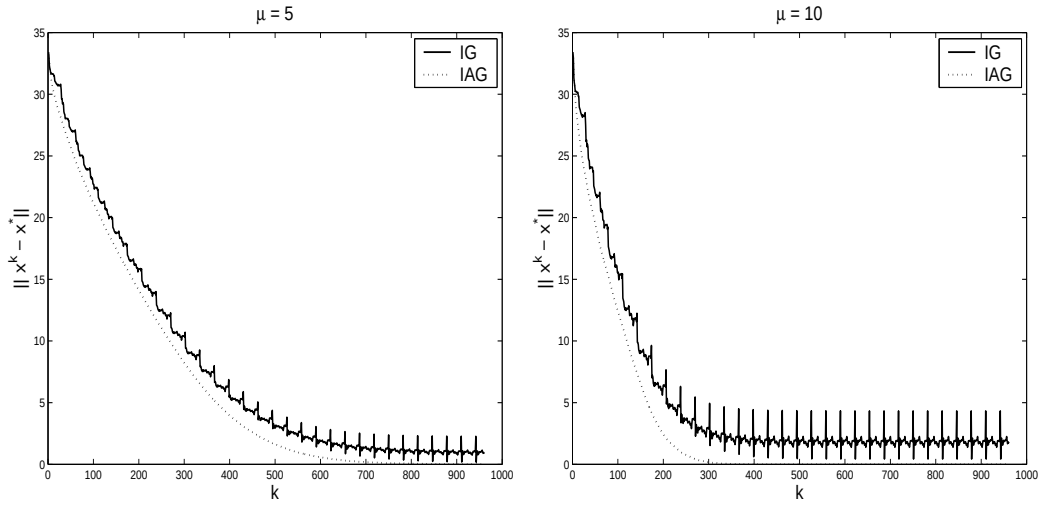

Figure 2.3: Distance of IG and IAG iterates to the optimal solution x^* for source localization problem.

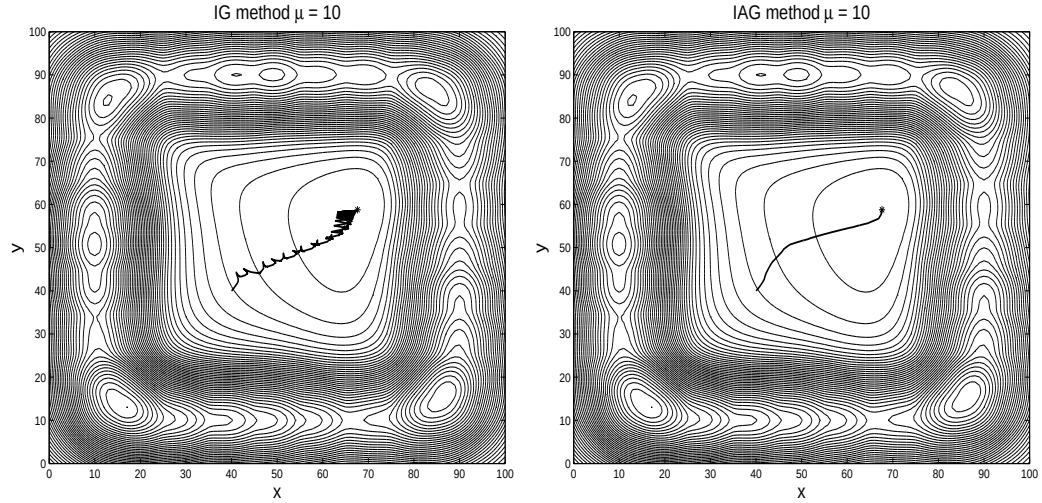

Figure 2.4: Path taken by the IG and IAG methods for source localization problem.

CHAPTER 3

On Tests for Global Maximum

3.1 Introduction

The maximum likelihood (ML) estimation method is one of the standard tools for parameter estimation. Among its appealing properties are consistency and asymptotic efficiency [55, 61, 115]. However, a major drawback of this method when applied to non-linear estimation problems is the fact that the associated likelihood equations required for the derivation of the estimator rarely have a closed form analytic solution. This shortcoming poses a global optimization problem. Solving this problem by applying numerical methods is usually computationally prohibitive. To date, there have been few global optimization methods applied to ML estimation (e.g. [4, 26, 106, 127, 128]) because of the computational complexity involved. More commonly, initiate and converge methods are applied. These methods are based on an initial guess (often found by a simpler method) which is followed by a local, often iterative, optimization procedure (e.g. the expectation maximization algorithm [38] and its variations [81], Fisher scoring [81], the Gauss-Newton method [110], and majorizing or minorizing algorithms [40], [56]). As a consequence, the performance of these methods highly depends on the starting point. In particular, if the log-

likelihood function is not strictly convex and there is no available method that is guaranteed to provide an initial guess within the attraction region of the global maximum, then there is a risk that a local search will stagnate at a local maximum. This phenomenon leads to large-scale estimation errors.

The maximum likelihood framework would benefit from an answer to the following question: Given a location of a relative maximum of the log-likelihood function, how to assess whether this is the global maximum? One approach to this question is the Kronecker-Picard integral framework [106]. However, the computation of this multi-dimensional integral is difficult, indeed equivalent to the complexity involved in finding the global maximum, rendering this approach impractical. Instead, in this paper we take a statistical approach to answering this question.

The first statistical solutions for discriminating between local and global maxima were based on sampling the domain of the log-likelihood function. Given a sequence of random starting points and the corresponding set of relative maxima found by a local search method, Finch et. al. [44] proposed a statistical method to assess the probability that the global maximum has not yet been found based on an asymptotic (in the number of starting points) result on the total probability of unobserved outcomes due to Bickel and Yahav [19]. Veall [118] used an order statistic result due to de Haan [53] that characterizes the distribution of the ordered values of a smooth function, sampled at random points. Given a relative maximum, the log-likelihood function is evaluated at a large number of randomly selected points. If a point with a value larger than the value of the candidate maximum is found, then clearly it is not the global maximum. If no such point is found, de Haan's result is used to assess the probability that the relative maximum is the global one. Since these methods are based on sampling the domain of the log-likelihood function, they suffers from the curse of dimensionality and do not generalize well to high dimensional problems. Yet

high dimensional problems are exactly those in which global optimization methods are computationally demanding.

Dorsey and Mayer [39] reported poor performance of Veall's method and, as an alternative, proposed to use the available methods for testing parametric models to answer the question at hand. They observed that a local maximum of the log-likelihood function is in fact a global maximum of a particular misspecified model - a model in which the parameters are restricted to a region that does not contain the true parameter. For scenarios in which the model is known to be correctly specified, these authors tested whether a relative maximum is the global one by applying a test that detects model mismatch. If the result of the test leads to the conclusion that a model mismatch is likely, the hypothesis that the relative maximum is the global one is rejected. Otherwise, the relative maximum is declared the final estimate. Independently, Gan and Jiang [49] made the same observation and proposed White's information matrix test [122] as a test for global maximum. More recently, Biernacki [20,21] proposed a new test, which is closely related to Cox's tests for separate families of hypotheses [33,34], and showed through simulations that his new test outperforms White's information matrix test.

A drawback of the methods of [39], [49], and [21] is that they are sensitive to model mismatch. In particular, when the model is not specified correctly, the tests lose their power to distinguish between local and global maxima. In some engineering applications the statistical model is derived from the underlying physical phenomenon and deviations from this model are unlikely. In these cases, the methods can be directly applied. However, when there are uncertainties about the model, the methods [39], [49], and [21] need to be modified so as to not classify a global maximum of a misspecified model as a local maximum.

In this paper, the tests are derived under possible model mismatch. The sensitiv-

ity of the tests to model mismatch is analyzed in terms of the Renyi divergence and the Kullback-Leibler distance between the true underlying distribution and the assumed parametric class. The analysis leads to a simple threshold correction method that accounts for possible deviations from the model as long as these deviations are bounded in terms of the mentioned distances. When deviations from the model are defined in terms of an embedding in a larger parametric class, insensitivity to a Pitman drift is established by constructing tests based on a vector valued validation function that is orthogonal to the elements of the gradient of the log-likelihood function of the larger class. This construction leads to tests that are locally robust to deviations from the assumed model.

An exhaustive catalogue of all the available methods for model specification testing that might be considered as candidates for tests for global maximum is beyond the scope of this paper. Rather, this paper focuses on the class of M-tests, which includes the tests of [49] and [21] as special cases, and investigates their performance as tests for global maximum.

The problem of testing a relative maximum is related to the problem of eliminating spurious maxima in scenarios in which the ML estimator (MLE) is not necessarily consistent or may not even exist (see [107] and references therein). Although some of the results apply to that problem as well, we do not pursue this connection here.

In Sec. 3.2, we review the properties of the MLE under a possible model mismatch and pose the problem of discriminating between local and global maxima as a statistical hypothesis testing problem. The general framework for constructing M-tests [88, 114, 123] is presented, and it is shown that two of the available tests in the literature are special cases of M-tests. In Sec. 3.3, the consistency of the tests is established and an approximation of the finite sample power of the tests is derived, which is useful for predicting performance and provides a measure for

comparing between tests. The problem of model mismatch is treated in Sec. 3.4. The effect of model mismatch is characterized in terms of the Renyi divergence and the Kullback-Leibler distance and two methods for making the tests robust to small deviations from the underlying model are given. Finally, to show the applicability of this framework, in Sec. 3.5 a Monte-Carlo evaluation of the performance of the tests is presented in terms of level and power under both correct and mismatched model.

3.2 Preliminaries

Let y_t, $t = 1, \ldots, n$ be a collection of n independent observations drawn from an unknown distribution G with density $g(y)$, $y \in \mathbb{R}^P$. The information we want to extract from the data is encoded in a $K \times 1$ parameter vector θ, through which we define a parametric family of densities $\{f(y, \theta) : \theta \in \Theta\}$ that are twice continuously differentiable in θ for all y. For scalar functions denote by $\nabla_\theta(\cdot)$ and $\nabla^2_\theta(\cdot)$ the column vector of partial derivatives and the Hessian matrix with respect to θ, respectively. For vector valued functions let $\nabla^T_\theta(\cdot)$ be the matrix whose (k, l) element is the partial derivative of the k'th element of the function with respect to the l'th element of θ. Assume that the elements of the matrices $\nabla_\theta \log f(y, \theta) \nabla^T_\theta \log f(y, \theta)$ and $\nabla^2_\theta \log f(y, \theta)$ are dominated by functions integrable with respect to G, for all $\theta \in \Theta$, a compact subspace of $\mathbb{R}^K$.

Denote by

$$L_n(Y_n; \theta) = \frac{1}{n} \sum_{t=1}^{n} \log f(y_t; \theta)$$

the normalized log-likelihood function of the measurements, where $Y_n = [y_1\, y_2\, \ldots\, y_n]$. The MLE[1] is defined as

$$\widehat{\theta}_n = \arg \max_{\theta \in \Theta} L_n(Y_n; \theta). \tag{3.1}$$

[1]Sometimes called quasi-MLE when the model is incorrect.

Denote by $\mathrm{E}\{\cdot\}$ the expectation with respect to the true underlying distribution G, and by θ^* the minimizer of the Kullback-Leibler information, i.e.,

$$\theta^* = \arg\min_{\theta\in\Theta} \mathrm{E}\left\{\log\frac{g(y)}{f(y;\theta)}\right\} = \arg\max_{\theta\in\Theta} a(\theta)$$

where $a(\theta)$ is the ambiguity function, defined as

$$a(\theta) = \mathrm{E}\{\log f(y;\theta)\} \tag{3.2}$$

and assume that θ^* is a well defined unique interior point of Θ. Define the matrices

$$\begin{aligned} A(\theta) &= \mathrm{E}\left\{\nabla_\theta^2 \log f(y;\theta)\right\} \\ B(\theta) &= \mathrm{E}\left\{\nabla_\theta \log f(y;\theta)\nabla_\theta^T \log f(y;\theta)\right\} \\ C(\theta) &= A^{-1}(\theta)B(\theta)A^{-1}(\theta) \end{aligned} \tag{3.3}$$

and assume that $A(\theta^*)$ and $B(\theta^*)$ are non-singular. Under these assumptions, Theorems 2.1, 2.2, and 3.2 of White [122] assert that

$$\widehat{\theta}_n \xrightarrow{a.s.} \theta^* \tag{3.4}$$

as $n \to \infty$, and $\widehat{\theta}_n$ is asymptotically Gaussian in the sense that

$$\sqrt{n}\left(\widehat{\theta}_n - \theta^*\right) \xrightarrow{D} N\left(0, C(\theta^*)\right). \tag{3.5}$$

When $g(y) = f(y,\theta^0)$ almost everywhere for some unique $\theta^0 \in \Theta$, we say that the model is correctly specified and this result becomes the standard consistency, and asymptotic Normality result for the MLE. More specifically, if the elements of the

matrix $\nabla_\theta^T \left[\nabla_\theta f(y,\theta) \cdot f(y,\theta)\right]$ are dominated by functions integrable with respect to ν, for all $\theta \in \Theta$, where ν is the dominating measure such that $g(y) = dG(y)/d\nu$, and the support of $f(y,\theta)$ does not depend on θ, then $C(\theta^0) = -A^{-1}(\theta^0) = B^{-1}(\theta^0)$ is the inverse of the Fisher information matrix (FIM) [115, p. 80].

Denote by $\widetilde{\theta}_n$ one of the relative maxima of the log-likelihood function. Then the problem addressed in this paper can be formulated as a hypothesis testing problem. Given $\widetilde{\theta}_n$, decide between

$$\begin{aligned} \mathrm{H}_0 : &\quad \widetilde{\theta}_n = \widehat{\theta}_n \\ \mathrm{H}_1 : &\quad \widetilde{\theta}_n \neq \widehat{\theta}_n. \end{aligned} \tag{3.6}$$

A statistical test which gives a solution to this problem is called a *test for global maximum*.

3.2.1 M-Tests for Global Maximum

M-tests were proposed in an econometric context by Newey [88], Tauchen [114], and White [123] as a general way of testing the validity of parametric models (see [124, Ch. 9] as well). The tests are based on a vector valued test function

$$e(y,\theta) : \mathbb{R}^P \times \Theta \to \mathbb{R}^Q \tag{3.7}$$

which is chosen to satisfy

$$\int e(y,\theta) f(y,\theta) dy = 0, \quad \forall \theta \in \Theta. \tag{3.8}$$

Hence, given the MLE $\widehat{\theta}_n$, large values of $1/n \sum_{t=1}^n e(y_t, \widehat{\theta}_n)$ indicate that a model mismatch is likely. Small values of $1/n \sum_{t=1}^n e(y_t, \widehat{\theta}_n)$ indicate that the model is

correctly specified or alternatively that the type of model mismatch is such that $g(y) \notin \{f(y,\theta) : \theta \in \Theta\}$ but

$$\int e(y,\theta^*)g(y)dy = 0. \tag{3.9}$$

The same framework can be used to construct tests for (3.6). First suppose that the model is correctly specified and that $e(y,\theta)$ is chosen to satisfy (3.8). Then, given a location of a relative maximum of the log-likelihood function $\widetilde{\theta}_n$, large values of $1/n\sum_{t=1}^{n} e(y_t,\widetilde{\theta}_n)$ indicate that it is not likely that $\widetilde{\theta}_n$ is the MLE. This directly extends to the case of model mismatch, if it is known that (3.9) holds.

The tests are constructed as follows. Assume that the elements of $e(y,\theta)$ are twice differentiable with respect to θ for every y, and that the elements of the vector $\nabla_\theta e(y,\theta)$ and the matrices $e(y,\theta)\nabla_\theta^T \log f(y,\theta)$ and $e(y,\theta)e^T(y,\theta)$ are dominated by functions integrable with respect to G for all $\theta \in \Theta$. Define the vectors

$$\begin{aligned} h_n(\theta) &= \frac{1}{n}\sum_{t=1}^{n} e(y_t,\theta) \\ h(\theta) &= \mathrm{E}\left\{e(y,\theta)\right\} \end{aligned} \tag{3.10}$$

and the $Q \times K$ matrices

$$\begin{aligned} H_n(\theta) &= \frac{1}{n}\sum_{t=1}^{n} \nabla_\theta^T e(y_t,\theta) \\ H(\theta) &= \mathrm{E}\left\{\nabla_\theta^T e(y,\theta)\right\}. \end{aligned} \tag{3.11}$$

Define the $Q \times Q$ matrix $V(\theta)$ by

$$\begin{aligned} \mathrm{E}\Big\{\big[e(y,\theta) - h(\theta) - H(\theta)A^{-1}(\theta)\nabla_\theta \log f(y;\theta)\big] \times \\ \big[e(y,\theta) - h(\theta) - H(\theta)A^{-1}(\theta)\nabla_\theta \log f(y;\theta)\big]^T\Big\} \end{aligned} \tag{3.12}$$

and its empirical estimate by

$$V_n(\theta) = \frac{1}{n}\sum_{t=1}^{n} \tag{3.13}$$
$$\left[e(y_t,\theta) - h_n(\theta) - H_n(\theta)A_n^{-1}(\theta)\nabla_\theta \log f(y_t;\theta)\right] \times$$
$$\left[e(y_t,\theta) - h_n(\theta) - H_n(\theta)A_n^{-1}(\theta)\nabla_\theta \log f(y_t;\theta)\right]^T$$

where

$$A_n(\theta) = \frac{1}{n}\sum_{t=1}^{n} \nabla_\theta^2 \log f(y_t;\theta) \tag{3.14}$$

and assume that $e(y,\theta)$ is such that $V(\theta^*)$ in (3.12) is nonsingular. Under the assumptions made above,

$$\sqrt{n}\left[h_n(\widehat{\theta}_n) - h(\theta^*)\right] \xrightarrow{D} N\left(0, V(\theta^*)\right) \tag{3.15}$$

$$V_n(\widehat{\theta}_n) \xrightarrow{a.s.} V(\theta^*) \tag{3.16}$$

element by element, $V_n(\widehat{\theta}_n)$ is nonsingular for sufficiently large n, and as a result,

$$n\left[h_n(\widehat{\theta}_n) - h(\theta^*)\right]^T V_n^{-1}(\widehat{\theta}_n)\left[h_n(\widehat{\theta}_n) - h(\theta^*)\right] \tag{3.17}$$

is asymptotically Chi-Squared distributed with Q degrees of freedom [88, 114, 123]. An elementary proof of this result is included in the Appendix for completeness.

Based on this result, tests for global maximum can be constructed as follows. Choose a function $e(y,\theta)$ having mean zero at the point θ^*, that is

$$h(\theta^*) = \mathrm{E}\left\{e(y,\theta^*)\right\} = 0. \tag{3.18}$$

The function $e(y,\theta)$ will be called the *global maximum validation function*. Under

H_0 and when (3.18) is satisfied, the statistic

$$S_n = n h_n^T(\widetilde{\theta}_n) V_n^{-1}(\widetilde{\theta}_n) h_n(\widetilde{\theta}_n) \tag{3.19}$$

with $V_n^{-1}(\widetilde{\theta}_n)$ computed by (3.13) is asymptotically Chi-Squared distributed with Q degrees of freedom, denoted by χ^2_Q. Denote by $F_{\chi^2_Q}(\cdot)$ the χ^2_Q cumulative distribution function. Therefore, a false alarm level α test of the hypotheses (3.6) is made by comparing S_n to $F^{-1}_{\chi^2_Q}(1-\alpha)$, which is the critical value of the χ^2_Q distribution for the desired false alarm level. If S_n exceeds the critical value, H_0 is rejected and one concludes that the iterative local search should be re-initiated in the hope of convergence to a different maximum. Otherwise, the null hypothesis cannot be rejected and $\widetilde{\theta}_n$ is declared the final estimate.

When the model is correctly specified, $\theta^* = \theta^0$ and Eq. (3.18) becomes

$$h(\theta^0) = \mathrm{E}\left\{e(y, \theta^0)\right\} = \int e(y, \theta^0) f(y, \theta^0) dy = 0. \tag{3.20}$$

A global maximum validation function $e(y, \theta)$ satisfying (3.20) can be constructed from any random function, e.g. call it $\overline{e}(y, \theta)$, by replacing it with the centered statistic:

$$e(y, \theta) = \overline{e}(y, \theta) - \int \overline{e}(y, \theta) f(y; \theta) dy. \tag{3.21}$$

This construction ensures that the mean of the validation function at the true parameter is zero. Under this construction, $h_n(\widetilde{\theta}_n)$ (3.10) becomes

$$h_n(\widetilde{\theta}_n) = \frac{1}{n} \sum_{t=1}^{n} \overline{e}(y_t, \widetilde{\theta}_n) - \int \overline{e}(y, \widetilde{\theta}_n) f(y; \widetilde{\theta}_n) dy \tag{3.22}$$

and the property $h(\theta^0) = \mathrm{E}\left\{e(y, \theta^0)\right\} = 0$ holds. This manipulation requires an analytical solution of the integral in (3.22) or its approximation via numerical inte-

gration.

Two tests for global maximum that are available in the literature fall into this framework. Taking $e(y,\theta)$ to be the vector valued function defined as

$$[e(y,\theta)]_q = \frac{\partial^2 \log f(y;\theta)}{\partial\theta_{i_q}\partial\theta_{j_q}} + \frac{\partial \log f(y;\theta)}{\partial\theta_{i_q}}\frac{\partial \log f(y;\theta)}{\partial\theta_{j_q}} \tag{3.23}$$

where $[\cdot]_q$ denotes the vector's q'th element, and the indices i_q and j_q, $q = 1,\ldots,Q$, are chosen so that $V(\theta^*)$ is nonsingular, we obtain White's information matrix test [122] which was used by Gan and Jiang as their test for global maximum [49]. This test is motivated by the fact that when the model is correctly specified, $A_n(\widehat{\theta}_n)$ defined in (3.14), and $B_n(\widehat{\theta}_n)$, defined by

$$B_n(\theta) = \frac{1}{n}\sum_{t=1}^{n} \nabla_\theta \log f(y_t;\theta)\nabla_\theta^T \log f(y_t;\theta) \tag{3.24}$$

converge a.s. as $n \to \infty$ to the -FIM and FIM, respectively; an idea that was originally used by White in his test for model mismatch [122]. Hence, when the model is correctly specified, (3.18) is satisfied since the expected value of the sum at θ^0 vanishes. Gan and Jiang noted that White's test suffers from slow convergence rates to unit power, i.e., it requires a large number of samples to detect local maxima with high probability. A test with better convergence rates was recently proposed by Biernacki [21]. The cost of this improvement is increased complexity due to the need to evaluate an integral of the type (3.22). The validation function $e(y,\theta)$ associated with Biernacki's test is the scalar function

$$e(y,\theta) = \log f(y;\theta) - \int \log f(y;\theta) f(y;\theta) dy \tag{3.25}$$

which is a special case of (3.21). Hence,

$$h_n(\widetilde{\theta}_n) = \frac{1}{n}\sum_{t=1}^{n} \log f(y_t;\widetilde{\theta}_n) - \int \log f(y;\widetilde{\theta}_n) f(y;\widetilde{\theta}_n) dy. \tag{3.26}$$

This test is closely related to Cox's tests of separate families of hypotheses [33], [34]. The choice (3.25) of $e(y,\theta)$ leads to a test that compares the log-likelihood evaluated at $\widetilde{\theta}_n$ to its expected value, which is calculated as if $\widetilde{\theta}_n$ is the true parameter. The test requires the evaluation of an integral (3.26) of dimension P - the dimension of y. This might be prohibitive in real time applications, although in Sec. 3.5.1 below, a closed form expression for the case of Gaussian distributed y_t is given. In [20, 21] the variance estimator required for the construction of S_n (3.19) is consistent for $\mathrm{E}\left\{e(y,\theta^0)e^T(y,\theta^0)\right\}$ rather than for $V(\theta^0)$ (3.12). From (3.28) below, it can be seen that under the null hypothesis H_0 and when the model is correctly specified, $\mathrm{E}\left\{e(y,\theta^0)e^T(y,\theta^0)\right\}$ is an upper bound on the asymptotic variance of $\sqrt{n}h_n(\widetilde{\theta}_n)$ (3.26). The bound is tight when either $B(\theta^0)$ is large, e.g., at high signal to noise ratio, or when $H(\theta^0)$ is small, i.e., the expectation of the gradient of $e(y,\theta)$ is small, but in general the variance estimator of [20, 21] leads to a test with a false alarm level smaller than the specified value.

3.2.2 Moments Matching Tests

Moments matching tests were previously proposed as tests for model mismatch (see e.g. [114]) but were not applied to the problem of discrimination of local maxima. The tests are based on the property that the moments of the distribution induced by the estimated parameter should be in good agreement with the empirical moments of the data. Therefore, these tests are especially suited for cases in which the underlying physical model specifies a simple parametrization of one of the moments of the data.

For example, assume that the mean of y is modelled by $\mu(\theta)$, i.e. $\mu(\theta) = \int y f(y;\theta)dy$, where $\mu(\cdot)$ is a pre-specified non-linear function, then to construct a test, which is based on the first moment, $e(y,\theta)$ is taken to be

$$e(y,\theta) = y - \mu(\theta).$$

This choice of $e(y,\theta)$ leads to the empirical estimate

$$h_n(\widetilde{\theta}_n) = \frac{1}{n}\sum_{t=1}^{n} y_t - \mu(\widetilde{\theta}_n).$$

It is clear that under a correctly specified model, equation (3.18) is satisfied. If the model is not correctly specified but the specification of the mean is correct, the condition

$$h(\theta^*) = \mathrm{E}\{y\} - \mu(\theta^*) = 0 \tag{3.27}$$

will still hold if the parametric class $\{f(y;\theta) : \theta \in \Theta\}$ belongs to the linear exponential family [124].

If the mean of the data does not depend on θ or is weakly dependent, one can improve the test by including higher order moments. For example, one can specify $e(y,\theta)$ as one or more elements of the difference between sample and ensemble covariance matrices:

$$[e(y,\theta)]_q = [y]_{i_q}[y]_{j_q} - [R(\theta)]_{i_q,j_q}, \quad q = 1,\ldots,Q$$

where for matrices $[\cdot]_{q,k}$ denotes the (q,k) element, and $[R(\theta)]_{i_q,j_q} = \int [y]_{i_q}[y]_{j_q} f(y;\theta)dy$ is pre-specified from the underlying model.

3.2.3 Covariance Matrix Estimation

It is possible to exploit properties of the null hypothesis H_0 (3.6) in order simplify and improve the estimator (3.13) of the covariance matrix of $\sqrt{n}h_n(\widetilde{\theta}_n)$ (see e.g. [49, 88, 114, 122, 124]). Under H_0 $\sqrt{n}h_n(\widetilde{\theta}_n)$ equals $\sqrt{n}h_n(\widehat{\theta}_n)$, and since by construction $h(\theta^*) = 0$, it is possible to drop the term $h_n(\widetilde{\theta}_n)$, which appears in (3.13) after substituting $\widetilde{\theta}_n$. Furthermore, when the model is correctly specified, under H_0, the asymptotic covariance matrix of $\sqrt{n}h_n(\widetilde{\theta}_n)$ simplifies to

$$\mathrm{E}\left\{e(y,\theta^0)e^T(y,\theta^0)\right\} - H(\theta^0)B^{-1}(\theta^0)H^T(\theta^0) \tag{3.28}$$

where $B(\theta)$ and $H(\theta)$ are given in (3.3) and (3.11), respectively, and since a correct model specification is assumed, expectations are taken with respect to the density $f(y,\theta^0)$. Using this property, the following covariance estimators can be considered. The first is based on the data and the form (3.28):

$$\begin{aligned}\widehat{V}_n(\widetilde{\theta}_n) = \frac{1}{n}\sum_{t=1}^{n} e(y_t,\widetilde{\theta}_n)e^T(y_t,\widetilde{\theta}_n) \\ - H_n(\widetilde{\theta}_n)B_n^{-1}(\widetilde{\theta}_n)H_n^T(\widetilde{\theta}_n)\end{aligned} \tag{3.29}$$

where $B_n(\theta)$ and $H_n(\theta)$ are defined in (3.24) and (3.11), respectively. In the correct model case, under H_0 the estimator (3.29) converges a.s. to the covariance matrix (3.28) [121, Lemma 3.1], and hence it is positive definite a.s. for sufficiently large n. The second estimator is given by

$$\begin{aligned}\overline{V}_n(\widetilde{\theta}_n) = \int e(y,\widetilde{\theta}_n)e^T(y,\widetilde{\theta}_n)f(y,\widetilde{\theta}_n)dy - \\ \overline{H}(\widetilde{\theta}_n)\overline{B}^{-1}(\widetilde{\theta}_n)\overline{H}^T(\widetilde{\theta}_n)\end{aligned} \tag{3.30}$$

where

$$\overline{B}(\theta) = \int \nabla_\theta \log f(y;\theta) \nabla_\theta^T \log f(y;\theta) f(y;\theta) dy$$

and

$$\overline{H}(\theta) = \int \nabla_\theta^T e(y,\theta) f(y;\theta) dy.$$

It should be noted that under H_1 or under model mismatch, these estimates are not necessarily consistent and the estimator (3.29) is not necessarily positive definite.

A number of authors investigated ways of estimating the covariance matrix in scenarios in which unexpected dependencies between the measurements may occur (see e.g. [124], [89] and references therein). Methods for eliminating the requirement for covariance matrix estimation altogether were recently proposed in [27] for the problem of model testing in non-linear regression.

3.3 Power Analysis

In order to derive the power function, the asymptotic distribution of $\widetilde{\theta}_n$ under H_1 needs to be determined. Therefore, assumptions on the structure of the ambiguity function (3.2) at different local maxima are required. Assume that the system of equations $\nabla a(\theta) = 0$, has a finite number of solutions in Θ and each one of these solutions is an interior point of Θ. In addition, at each of these points, the matrix $\nabla^2 a(\theta)$ is either negative definite or positive definite. The ambiguity function $a(\theta)$ has its global maximum at θ^*; denote by θ^m, $m = 1, \ldots, M$, the other M local maxima of $a(\theta)$.

Theorem 1. *For sufficiently large n, $L_n(Y_n;\theta)$ has $M+1$ local maxima for almost every sequence $\{y_t\}_{t\geq 1}$. Furthermore, the location of these relative maxima are strongly consistent estimates for θ^* and θ^m, $m = 1, \ldots, M$.*

Proof. The outline of the proof goes as follows. First we prove that, for sufficiently large n, the norm of the first derivative vector of $L_n(Y_n;\theta)$ is strictly positive outside of arbitrary small neighborhoods of the local maxima and local minima of $a(\theta)$. Then, we prove that when restricted to these neighborhoods, $L_n(Y_n;\theta)$ is either strictly convex or strictly concave and hence has a single minimum or a single maximum, respectively.

Under the assumptions made, [57, Thm. 2] gives the following uniform strong law of large numbers:

$$\begin{aligned} L_n(Y_n;\theta) &\to \mathrm{E}\left\{\log f(y;\theta)\right\} \\ \nabla_\theta L_n(Y_n;\theta) &\to \mathrm{E}\left\{\nabla_\theta \log f(y;\theta)\right\} \\ \nabla^2_\theta L_n(Y_n;\theta) &\to \mathrm{E}\left\{\nabla^2_\theta \log f(y;\theta)\right\} \end{aligned} \tag{3.31}$$

as $n \to \infty$ uniformly in Θ for almost every sequence $\{y_t\}_{t\geq 1}$.

Denote the relative minimum points for the ambiguity function by $\phi^j \in \Theta$, $j = 1,\ldots,J$, $J \geq 0$. By the assumption, $\nabla_\theta a(\theta) = 0$ at the points θ^*, $\theta^m, m = 1,\ldots,M$ and $\phi^j, j = 1,\ldots,J$ and only at these points. In addition, the matrix $\nabla^2 a(\theta)$ is negative definite at the points θ^*, $\theta^m, m = 1,\ldots,M$ and positive definite at the points $\phi^j, j = 1,\ldots,J$. Denote the eigenvalues of the matrix $\nabla^2 a(\theta)$ by $\lambda_k(\theta), k = 1,\ldots,K$. Therefore,

$$\begin{aligned} &\max_k\{\lambda_k(\theta^*)\} < 0 \\ &\max_k\{\lambda_k(\theta^m)\} < 0, \quad \forall m = 1,\ldots,M \end{aligned}$$

and

$$\min_k\{\lambda_k(\phi^j)\} > 0, \quad \forall j = 1,\ldots,J.$$

The eigenvalues are continuous functions of the matrix element and the operations max and min are also continuous in their arguments. Therefore, there are disjoint open neighborhoods $\mathcal{N}^*$, $\mathcal{N}^m$, and $\mathcal{M}^j$ around θ^*, θ^m and ϕ^j, respectively, $m = 1, \ldots, M$, $j = 1, \ldots J$, that satisfy the following conditions:

$$\begin{aligned} &\sup_{\theta \in \mathcal{N}^*} \max_k \{\lambda_k(\theta)\} \leq \overline{\delta} < 0 \\ &\sup_{\theta \in \mathcal{N}^m} \max_k \{\lambda_k(\theta)\} \leq \overline{\delta} < 0, \quad \forall m = 1, \ldots, M \\ &\inf_{\theta \in \mathcal{M}^j} \min_k \{\lambda_k(\theta)\} \geq \underline{\delta} > 0, \quad \forall j = 1, \ldots, J. \end{aligned} \tag{3.32}$$

Denote

$$\widetilde{\Theta} = \Theta \setminus \left[\mathcal{N}^* \bigcup \left(\bigcup_{m=1}^{M} \mathcal{N}^m \right) \bigcup \left(\bigcup_{j=1}^{J} \mathcal{M}^j \right) \right].$$

Since $\widetilde{\Theta}$ is also compact, and $|\partial a(\theta)/\partial \theta_k|$ is bounded and continuous for all k, we have

$$\inf_{\theta \in \widetilde{\Theta}} \sum_{k=1}^{K} |\partial a(\theta)/\partial \theta_k| = \min_{\theta \in \widetilde{\Theta}} \sum_{k=1}^{K} |\partial a(\theta)/\partial \theta_k| = \delta.$$

Since by the assumption all the stationary points of $a(\theta)$ are outside of $\widetilde{\Theta}$, δ is strictly positive.

Next, we prove that there exist N_1 such that $\forall n > N_1$,

$$\sum_{k=1}^{K} |\partial L_n(Y_n; \theta)/\partial \theta_k| > \delta/2, \quad \forall \theta \in \widetilde{\Theta}, \quad w.p.\ 1$$

i.e., for sufficiently large n, the function $L_n(Y_n; \theta)$ has no stationary points in $\widetilde{\Theta}$ for almost every sequence $\{y_t\}_{t \geq 1}$. To this end, choose N_1 such that for all $n > N_1$,

$$\begin{aligned} &|\partial a(\theta)/\partial \theta_k - \partial L_n(Y_n; \theta)/\partial \theta_k| < \frac{\delta}{2K}, \\ &\forall k = 1, \ldots, K, \forall \theta \in \widetilde{\Theta}, \quad w.p.\ 1 \end{aligned}$$

which can always be found by (3.31). Therefore,

$$\sum_{k=1}^{K} |\partial a(\theta)/\partial\theta_k - \partial L_n(Y_n;\theta)/\partial\theta_k| < \frac{\delta}{2},$$
$$\forall\theta \in \widetilde{\Theta}, \quad w.p.\ 1$$

and hence, $\forall n > N_1$,

$$\sum_{k=1}^{K} |\partial L_n(Y_n;\theta)/\partial\theta_k| > \frac{\delta}{2}, \quad \forall\theta \in \widetilde{\Theta}, \quad w.p.\ 1$$

and the claim is proved.

Next, we prove that there exist N_2 such that $\forall n > N_2$, $L_n(Y_n;\theta)$ is concave over $\overline{\mathcal{N}}^*$, $\overline{\mathcal{N}}^m, m = 1,\ldots,M$ and convex over $\overline{\mathcal{M}}^j, j = 1,\ldots,J$, where $\overline{\mathcal{N}}$ denotes the closure of the set $\mathcal{N}$. Denote the eigenvalues of $\nabla^2 L_n(Y_n;\theta)$ by $\lambda_k^n(\theta), k = 1,\ldots,L$. We consider one specific neighborhood $\overline{\mathcal{N}}^*$, and prove that

$$\max_{\theta\in\overline{\mathcal{N}}^*} \max_k \{\lambda_k^n(\theta)\} < \frac{\overline{\delta}}{2} < 0, \quad \forall n > N_2, \quad w.p.\ 1 \tag{3.33}$$

where $\overline{\delta}$ was defined in (3.32), i.e., $L_n(Y_n;\theta)$ is concave over $\overline{\mathcal{N}}^*$.

By the construction, the maximal eigenvalue is uniformly continuous over $\overline{\mathcal{N}}^*$. Therefore,

$$\max_k \{\lambda_k^n(\theta)\} \to \max_k \{\lambda_k(\theta)\}, \quad \forall\theta \in \overline{\mathcal{N}}^*, \quad w.p.\ 1$$

and (3.33) follows. The same argument holds for the proof of concavity of $L_n(Y_n;\theta)$ over the rest of the neighborhoods $\overline{\mathcal{N}}^m$, $m = 1,2,\ldots,M$ and the convexity of $L_n(Y_n;\theta)$ over $\overline{\mathcal{M}}^j$, $j = 1,\ldots,J$.

For each set $\overline{\mathcal{N}}^m$, by (3.31) as n increases $L_n(Y_n;\theta)$ will eventually be greater at θ^m than at any point on the boundary of $\overline{\mathcal{N}}^m$, $w.p.\ 1$. Therefore, $L_n(Y_n;\theta)$ will

attain a single local maximum at an interior point of $\mathcal{N}^m$, *w.p.* 1 (not necessarily at θ^m). A similar argument holds for $\overline{\mathcal{N}}^*$ and for a minimum point in $\mathcal{M}^j$ and the first part of the theorem is proved.

Finally, since the sets $\mathcal{N}^*$, $\mathcal{N}^m, m = 1, \ldots, M$ can be taken arbitrarily small, the maximum points of $L_n(Y_n; \theta)$ are strongly consistent estimates of θ^*, $\theta^m, m = 1, \ldots, M$. □

Theorem 1 ensures that as n increases the relative maxima of the log-likelihood function occur close to the relative maxima of the ambiguity function and only at these locations. This implies that the relative maxima of the log-likelihood function are asymptotically Gaussian distributed. More specifically, let Θ^m be a closed neighborhood of θ^m, in which θ^m is the highest relative maximum of $a(\theta)$. Define the m'th local-MLE by

$$\widehat{\theta}_n^m = \arg \max_{\theta \in \Theta^m} L_n(Y_n; \theta), \quad m = 1, \ldots, M. \tag{3.34}$$

If the optimization method used to solve (3.1) is certain to find a relative maximum of $L_n(Y_n; \theta)$, then Theorem 1 asserts that for sufficiently large n, $\widetilde{\theta}_n$ will be equal to one of the local-MLEs $\widehat{\theta}_n^m$, w.p. 1. The local-MLE $\widehat{\theta}_n^m$ is the MLE associated with the model $\{f(y, \theta) : \theta \in \Theta^m\}$ and therefore falls into the mismatch model framework of White [122]. Hence we have the following.

Corollary 1. *For all m:*

1. $\widehat{\theta}_n^m \stackrel{a.s.}{\to} \theta^m$ *as* $n \to \infty$, *and*
2. $\sqrt{n}\left(\widehat{\theta}_n^m - \theta^m\right) \stackrel{D}{\to} N\left(0, C(\theta^m)\right)$.

In addition, by (3.15)-(3.17) we obtain the following:

Corollary 2. *For all m:*

$$\sqrt{n}\left[h_n(\widehat{\theta}_n^m) - h(\theta^m)\right] \xrightarrow{D} N\left(0, V(\theta^m)\right)$$

$V_n(\widehat{\theta}_n^m) \xrightarrow{a.s.} V(\theta^m)$ *element by element. In addition, assuming that* $V(\theta^m)$ *is nonsingular,*

$$n\left[h_n(\widehat{\theta}_n^m) - h(\theta^m)\right]^T V_n^{-1}(\widehat{\theta}_n^m)\left[h_n(\widehat{\theta}_n^m) - h(\theta^m)\right] \tag{3.35}$$

is asymptotically distributed as χ^2_Q.

From Corollary 2 it is clear that for the test to have power against $\widehat{\theta}_n^m$, $h(\theta^m)$ must not equal 0. Otherwise the statistic has the same asymptotic χ^2_Q distribution under both hypotheses H_0 and H_1 (3.6). On the other hand, if $h(\theta^m) \neq 0$ the consistency of the test can be established.

Corollary 3. *Assume* $\widetilde{\theta}_n = \widehat{\theta}_n^m$. *If* $h(\theta^m) \neq 0$ *then*

$$\Pr\{S_n > F^{-1}_{\chi^2_Q}(1-\alpha)\} \to 1$$

for every choice of level $\alpha \in (0,1)$.

Proof. Under the assumption, $h_n(\widetilde{\theta}_n) \xrightarrow{a.s.} h(\theta^m)$ by [121, Lemma 3.1]. Therefore, since $V_n(\widehat{\theta}_n^m) \xrightarrow{a.s.} V(\theta^m)$ element by element and we assumed that $V(\theta^m)$ is nonsingular,

$$\Pr\{S_n > \varepsilon\} \to 1$$

for all $\varepsilon > 0$, by [124, Thm. 8.13]. □

Implied from corollary 3 is the consistency of the test: If $h(\theta^m) \neq 0$ for all $m = 1, \ldots, M$, then

$$\Pr\{S_n > F^{-1}_{\chi^2_Q}(1-\alpha) | \mathrm{H}_1\} \to 1 \tag{3.36}$$

for every choice of level $\alpha \in (0,1)$, i.e., the test is consistent. This result extends the results of [49] and [21], which established under a correctly specified model (each for their own global maximum validation function) that if the only solution to the set of equations

$$\int \nabla_\theta \log f(y,\theta) f(y,\theta^0) dy = 0$$
$$\int e(y,\theta) f(y,\theta^0) dy = 0$$

is θ^0, then

$$\sqrt{n} h_n(\widetilde{\theta}_n) \xrightarrow{D} N(0, V(\theta^0)) \quad \text{iff} \quad \widetilde{\theta}_n = \widehat{\theta}_n.$$

Furthermore, Corollary 2 implies that under H_1, and particularly when $\widetilde{\theta}_n = \widehat{\theta}_n^m$, the distribution of the test statistic S_n is approximately non-central χ^2_Q with non-centrality parameter

$$n\delta^m = n h^T(\theta^m) V^{-1}(\theta^m) h(\theta^m)$$

denoted by $\chi^2_Q(n\delta^m)$ [58]. We denote the $\chi^2_Q(n\delta^m)$ cumulative distribution function by $F_{\chi^2_Q(n\delta^m)}(\cdot)$. The finite sample power of the test against a local maximum at θ^m can be approximated by [58, p. 468]

$$1 - F_{\chi^2_Q(n\delta^m)}\left[F^{-1}_{\chi^2_Q}(1-\alpha)\right]. \tag{3.37}$$

Therefore, the power of a given test against a local maximum at θ^m is characterized by

$$\delta^m = h^T(\theta^m) V^{-1}(\theta^m) h(\theta^m) \tag{3.38}$$

which will be called the power characteristic of the test as a function of θ^m. The power characteristic is a basis of comparison between tests.

3.4 Misspecified Models

In general, it is difficult to discriminate between the cases of: (a) $\widetilde{\theta}_n$ a local maximum in a correctly specified model; and (b) $\widetilde{\theta}_n$ a global maximum in a misspecified model. Under model mismatch, the probability of mistakenly rejecting $\widetilde{\theta}_n$ as the global maximum, increases with the number of samples.

If the test statistic is designed under the assumption that the model is correctly specified but the actual underlying distribution is outside the assumed parametric family, then (3.18) may be violated. In this case, even when $\widetilde{\theta}_n = \widehat{\theta}_n$, $h_n(\widetilde{\theta}_n) \overset{a.s.}{\to} h(\theta^*) \neq 0$ and, similar to the discussion in the previous section, S_n is approximately distributed as $\chi^2_Q(n\epsilon)$ with non-centrality parameter $n\epsilon = nh^T(\theta^*)V^{-1}(\theta^*)h(\theta^*)$, instead of the assumed central chi-squared. In this case, as n tends to infinity, the probability of mistakenly rejecting $\widetilde{\theta}_n$ as the global maximum increases to one regardless of the test threshold, and is approximately given by

$$1 - F_{\chi^2_Q(n\epsilon)}\left[F^{-1}_{\chi^2_Q}(1-\alpha)\right].$$

3.4.1 A Bound on the Non-Centrality Parameter

It is possible to bound the non-centrality parameter ϵ, induced by the model mismatch, in terms of the Renyi divergence between $f(y;\theta^*)$ and true underlying density $g(y)$. Consider the case in which $e(y,\theta)$ is a scalar function and satisfies

$$\int e(y,\theta)f(y,\theta)dy = 0, \quad \forall\theta \in \Theta.$$

In this case the non-centrality parameter simplifies to

$$n\epsilon = nh^2(\theta^*)/V(\theta^*).$$

Since θ^* minimizes $D\left(g(y)||f(y,\theta)\right)$ with respect to θ,

$$\int \nabla_\theta \log f(y,\theta)|_{\theta=\theta^*}\, g(y)dy = 0.$$

Therefore, denoting

$$d(y,\theta) = e(y,\theta) - H(\theta)A^{-1}(\theta)\nabla_\theta^T \log f(y,\theta)$$

we obtain

$$\begin{aligned} h(\theta^*) &= \mathrm{E}\left\{e(y,\theta^*)\right\} = \mathrm{E}\left\{d(y,\theta^*)\right\} \\ &= \int \left[d(y,\theta^*) - h(\theta^*)\right]\left[g(y) - f(y,\theta^*)\right] dy. \end{aligned}$$

By the Cauchy-Schwartz inequality

$$\begin{aligned} h^2(\theta^*) \leq & \int \left[d(y,\theta^*) - h(\theta^*)\right]^2 g(y)dy \times \\ & \int \frac{\left[g(y) - f(y,\theta^*)\right]^2}{g(y)} dy \\ = & V(\theta^*)\left(\int \frac{f^2(y,\theta^*)}{g(y)} dy - 1\right) \end{aligned}$$

implying that

$$\epsilon = \frac{h^2(\theta^*)}{V(\theta^*)} \leq \exp\left[D_2\left(f(y,\theta^*)||g(y)\right)\right] - 1$$

where

$$D_\alpha(f_1(y)||f_2(y)) = \frac{1}{\alpha - 1}\log \int f_1^\alpha(y) f_2^{1-\alpha}(y)dy$$

is the Renyi divergence between $f_1(y)$ and $f_2(y)$ with parameter α.

Therefore, when a bound on $D_2\left(f(y,\theta^*)||g(y)\right)$ is available, say B_ϵ, it is possible to set the threshold of the test according to a $\chi^2_Q(n\left[\exp(B_\epsilon) - 1\right])$ distribution, i.e.,

reject the null hypothesis if

$$S_n > F^{-1}_{\chi^2_Q(n[\exp(B_\epsilon)-1])}(1-\alpha). \tag{3.39}$$

This choice of threshold leads to a test, the level of which decreases to zero, instead of increasing to one. Since

$$F^{-1}_{\chi^2_Q(n[\exp(B_\epsilon)-1])}(1-\alpha) > F^{-1}_{\chi^2_Q}(1-\alpha)$$

for all α [58], this adjustment decreases the power of the test. However, as long as the the power characteristic of the test at a local maximum δ^m (3.38) is larger than $\exp(B_\epsilon)-1$, the test will detect such a local maximum with probability approaching one as n tends to infinity.

Often it is difficult to compute a bound on $D_2\left(f(y,\theta^*)||g(y)\right)$, especially due to the computation required for θ^*. When the true underlying distribution and the assumed parametric model are both embedded in a larger parametric class and are sufficiently close to one another, it is possible to approximate the Renyi divergence by the Kullback-Leibler distance defined below. This leads to a simple approximation of B_ϵ.

Suppose that the parametric class $\{f(y;\theta) : \theta \in \Theta\}$ is embedded in a larger class $\{\widetilde{f}(y;\theta,\gamma) : \theta \in \Theta, \gamma \in \Gamma \subset \mathbb{R}^{K'}\}$ such that $f(y;\theta) = \widetilde{f}(y;\theta,\gamma^0)$ for all $\theta \in \Theta$, and that the true underlying density is $g(y) = \widetilde{f}(y;\theta^0,\gamma^1)$, with θ^1 close to θ^0. This setting was recently treated in [126], where the parameter vector γ was referred to as the background parameter.

In this case, the local equivalence and symmetry of f-divergence measures [3, p.

85] can be used to approximate the Renyi divergence

$$D_2\left(f(y,\theta^*)||g(y)\right) = D_2\left(\widetilde{f}(y;\theta^*,\gamma^0)||\widetilde{f}(y;\theta^0,\gamma^1)\right)$$

by

$$2D_1\left(\widetilde{f}(y;\theta^0,\gamma^1)||\widetilde{f}(y;\theta^*,\gamma^0)\right)$$

up to terms of order $O\left(||\theta^*-\theta^0||^3+||\gamma^0-\gamma^1||^3\right)$, where

$$\begin{aligned} D_1\left(f_1(y)||f_2(y)\right) &= \lim_{\alpha\to 1} D_\alpha\left(f_1(y)||f_2(y)\right) \\ &= \int \log\left(\frac{f_1(y)}{f_2(y)}\right) f_1(y)dy \end{aligned}$$

is the Kullback-Leibler distance between $f_1(y)$ and $f_2(y)$.

Furthermore, θ^* minimizes $D_1\left(\widetilde{f}(y;\theta^0,\gamma^1)||\widetilde{f}(y;\theta,\gamma^0)\right)$ over $\theta\in\Theta$. Hence,

$$\begin{aligned} &D_1\left(\widetilde{f}(y;\theta^0,\gamma^1)||\widetilde{f}(y;\theta^*,\gamma^0)\right) \leq \\ &\qquad D_1\left(\widetilde{f}(y;\theta^0,\gamma^1)||\widetilde{f}(y;\theta^0,\gamma^0)\right). \end{aligned}$$

Therefore, $D_2\left(f(y,\theta^*)||g(y)\right)$ can be bounded by $2D_1\left(\widetilde{f}(y;\theta^0,\gamma^1)||\widetilde{f}(y;\theta^0,\gamma^0)\right)$ up to terms of order $O\left(||\theta^*-\theta^0||^3+||\gamma^0-\gamma^1||^3\right)$. The advantage of the bound is that it does not require the difficult evaluation of θ^*.

3.4.2 Tests Insensitive to a Pitman Drift

Assume again that the parametric class $\{f(y;\theta):\theta\in\Theta\}$ is embedded in a larger class $\{\widetilde{f}(y;\theta,\gamma):\theta\in\Theta,\ \gamma\in\Gamma\subset\mathbb{R}^{K'}\}$ such that $f(y;\theta)=\widetilde{f}(y;\theta,\gamma^0)$ for all $\theta\in\Theta$. Denote by $\beta=[\theta^T,\gamma^T]^T$ the concatenated parameter vector and assume that there exist integrable functions $a(y)$ and $b(y)$ such that $a(y)b(y)$ is integrable as well with

respect to ν, and for almost all y, $\widetilde{f}(y;\beta) \leq a(y)$ and $|\log \widetilde{f}(y;\beta)|$, $|\nabla_\beta \log \widetilde{f}(y;\beta)|^2$, $|\nabla^2_\beta \log \widetilde{f}(y;\beta)|$, $|e(y,\theta)|^2$, and $|\nabla_\theta e(y,\theta)|$ are each less than $b(y)$ for all $\beta \in \Theta \times \Gamma$, where for matrices $|\cdot|$ denotes the maximum valued element. Furthermore, assume that the support of $\widetilde{f}(y;\beta)$ in independent of β. Assume that the true underlying distribution depends on n, hence denoted by $g_n(y)$, and is given by

$$g_n(y) = \widetilde{f}(y;\theta^0, \gamma^0 + \gamma/\sqrt{n}) \tag{3.40}$$

for some fixed $\gamma \in \Gamma$, and denote the limiting distribution by $g(y)$. In the context of model specification tests, this type of local alternative is called a Pitman drift. Newey [88] investigated the power of M-tests to such local alternatives. Applying Newey's result to our setting we obtain that if $e(y,\theta)$ satisfies

$$\int e(y,\theta) f(y;\theta) dy = 0, \quad \forall \theta \in \Theta$$

then under H_0,

$$\sqrt{n} h_n(\widetilde{\theta}_n) \xrightarrow{D} N\left(D\gamma, V(\theta^0)\right) \tag{3.41}$$

where in the definition of $V(\theta)$ (3.12), the expectation is taken with respect to the density $f(y,\theta^0)$ and the term $h(\theta^0)$ vanishes. The term D in (3.41) is

$$\begin{aligned} D = &\int e(y,\theta^0)\, \nabla^T_\gamma \log \widetilde{f}(y;\theta^0,\gamma)\Big|_{\gamma=\gamma^0} f(y;\theta^0) dy \\ &- H(\theta^0) A^{-1}(\theta^0) \widetilde{B}_{(\theta,\gamma)}(\beta^0) \end{aligned}$$

where the expectations in the definition of $A(\theta)$ and $H(\theta)$, (3.3) and (3.11), respectively, are taken with respect to the density $f(y,\theta^0)$ as well. $\beta^0 = [\theta^{0T}, \gamma^{0T}]^T$ and the matrix $\widetilde{B}_{(\theta,\gamma)}(\beta)$ is the upper right $K \times K'$ block of the FIM associated with the

density $\widetilde{f}(y;\beta)$, that is,

$$\widetilde{B}(\beta) = \int \nabla_\beta \log \widetilde{f}(y;\beta) \nabla_\beta^T \log \widetilde{f}(y;\beta) \widetilde{f}(y;\beta) dy, \tag{3.42}$$

and it is assumed that $\widetilde{B}(\beta)$ is non-singular for all $\beta \in \Theta \times \Gamma$. Hence, S_n, defined in (3.19), is asymptotically non-central chi-squared distributed with Q degrees of freedom and non-centrality parameter

$$\delta = \gamma' D' V^{-1}(\theta^0) D\gamma.$$

In [88] this result is used to assess and optimize the power of M-tests against local alternatives. Here, our goal is reversed; we would like the tests to be insensitive to small deviations from the assumed model. Specifically, note that

$$\begin{aligned} H(\theta^0) &= \int \nabla_\theta \, e(y,\theta)|_{\theta=\theta^0} \, f(y;\theta^0) dy \\ &= -\int e(y,\theta) \, \nabla_\theta^T \log \widetilde{f}(y;\theta,\gamma^0)\Big|_{\theta=\theta^0} \, f(y;\theta^0) dy. \end{aligned}$$

Therefore, considering the space of zero-mean L_2 functions of y with inner product

$$\langle f_1(y), f_2(y) \rangle = \int f_1(y) f_2(y) f(y;\theta) dy$$

our objective is to construct a global maximum validation function $e(y,\theta)$, with elements orthogonal to the space spanned by the $K + K'$ set of functions

$$\nabla_\beta \log \widetilde{f}(y;\beta)\Big|_{\gamma=\gamma^0}. \tag{3.43}$$

By this construction, both terms of the matrix D are zeroed out, i.e., the test is

insensitive to the Pitman drift regardless of the vector γ. Denoting the classes of log-likelihood functions $\{\log f(y;\theta) : \theta \in \Theta\}$ and $\{\log \widetilde{f}(y;\theta,\gamma) : \theta \in \Theta,\, \gamma \in \Gamma\}$ by $\mathcal{F}$ and $\mathcal{G}$, respectively, Fig. 3.1 gives a geometrical interpretation of the construction of $e^{\perp}(y,\theta)$.

Given any global maximum validation function $e(y,\theta)$ that satisfies

$$\int e(y,\theta) f(y;\theta) dy = 0,\ \forall \theta \in \Theta,$$

its orthogonal component with respect to the vector (3.43), denoted by $e^{\perp}(y,\theta)$, is

$$e^{\perp}(y,\theta) = e(y,\theta) - \left[E(\beta) \widetilde{B}^{-1}(\beta) \nabla_{\beta} \log \widetilde{f}(y;\beta) \right]_{\gamma=\gamma^0} \tag{3.44}$$

where $E(\beta)$ is the $K \times (K + K')$ matrix of inner products between the elements of $e(y,\theta)$ and the functions in (3.43), given by

$$E(\beta) = \int e(y,\theta) \nabla_{\beta}^{T} \log \widetilde{f}(y;\beta) f(y;\theta) dy. \tag{3.45}$$

This can be verified by computing the matrix

$$\int e^{\perp}(y,\theta)\ \nabla_{\beta}^{T} \log \widetilde{f}(y;\beta) \Big|_{\gamma=\gamma^0} f(y;\theta) dy.$$

At any local maximum $\widetilde{\theta}_n$, $\sum_{t=1}^{n} \nabla_{\theta} \log f(y_t;\widetilde{\theta}_n) = 0$ and therefore, computing $h_n^{\perp}(\widetilde{\theta}_n) = \sum_{t=1}^{n} e^{\perp}(y_t,\widetilde{\theta}_n)$ reduces to

$$h_n^{\perp}(\widetilde{\theta}_n) = \sum_{t=1}^{n} e(y_t,\widetilde{\theta}_n) - E\,(\beta)\, \widetilde{B}_2(\beta) \sum_{t=1}^{n} \nabla_{\gamma} \log \widetilde{f}(y_t;\beta) \Big|_{\theta=\widetilde{\theta}_n, \gamma=\gamma^0}$$

where $\widetilde{B}_2(\beta)$ is the $(K+K')\times K'$ matrix composed of the right K' columns of $\widetilde{B}^{-1}(\beta)$ defined in (3.42). Furthermore, under the null hypothesis H_0, a consistent estimator for the covariance matrix of $\sqrt{n}h_n^{\perp}(\widetilde{\theta}_n)$ is

$$\frac{1}{n}\sum_{t=1}^{n} e^{\perp}\left(y_t,\widetilde{\theta}_n\right) e^{\perp}\left(y_t,\widetilde{\theta}_n\right)^T$$

since the term $H(\theta)$ (3.11), which appears in (3.28), is zero by construction of $e^{\perp}(y,\theta)$. When closed form expressions for $E(\beta)$ and $B(\beta)$ are available, the covariance matrix can also be consistently estimated under H_0 by

$$\begin{aligned}\widetilde{V}_n(\widetilde{\theta}_n) = \int e(y,\widetilde{\theta}_n)e^T(y,\widetilde{\theta}_n)f(y,\widetilde{\theta}_n)dy - \\ E(\widetilde{\theta}_n,\gamma^0)B^{-1}(\widetilde{\theta}_n,\gamma^0)E^T(\widetilde{\theta}_n,\gamma^0).\end{aligned} \tag{3.46}$$

In summary, tests for global maximum which are based on $e^{\perp}(y,\theta)$ are locally insensitive to model mismatch of the type defined in (3.40) for any $\gamma\in\Gamma$.

Another motivation for using $e^{\perp}(y,\theta)$ can be obtained from the Taylor expansion of $h(\theta^*)$ around γ^0. Assuming the derivatives can be taken inside the integrals, we obtain that the zeroth order (constant) term is identically zero and the first order (linear) term is zeroed by the construction of $e^{\perp}(y,\theta)$.

In practice, we expect these tests to be less sensitive to small deviations from the model. An example in which this is the case is given in Sec. 3.5.1.

3.5 Applications

The asymptotic regime adopted throughout the paper, raises the question of small sample performance. In this section, tests for global maximum will be derived and evaluated through simulations for several parameter estimation problems. In the

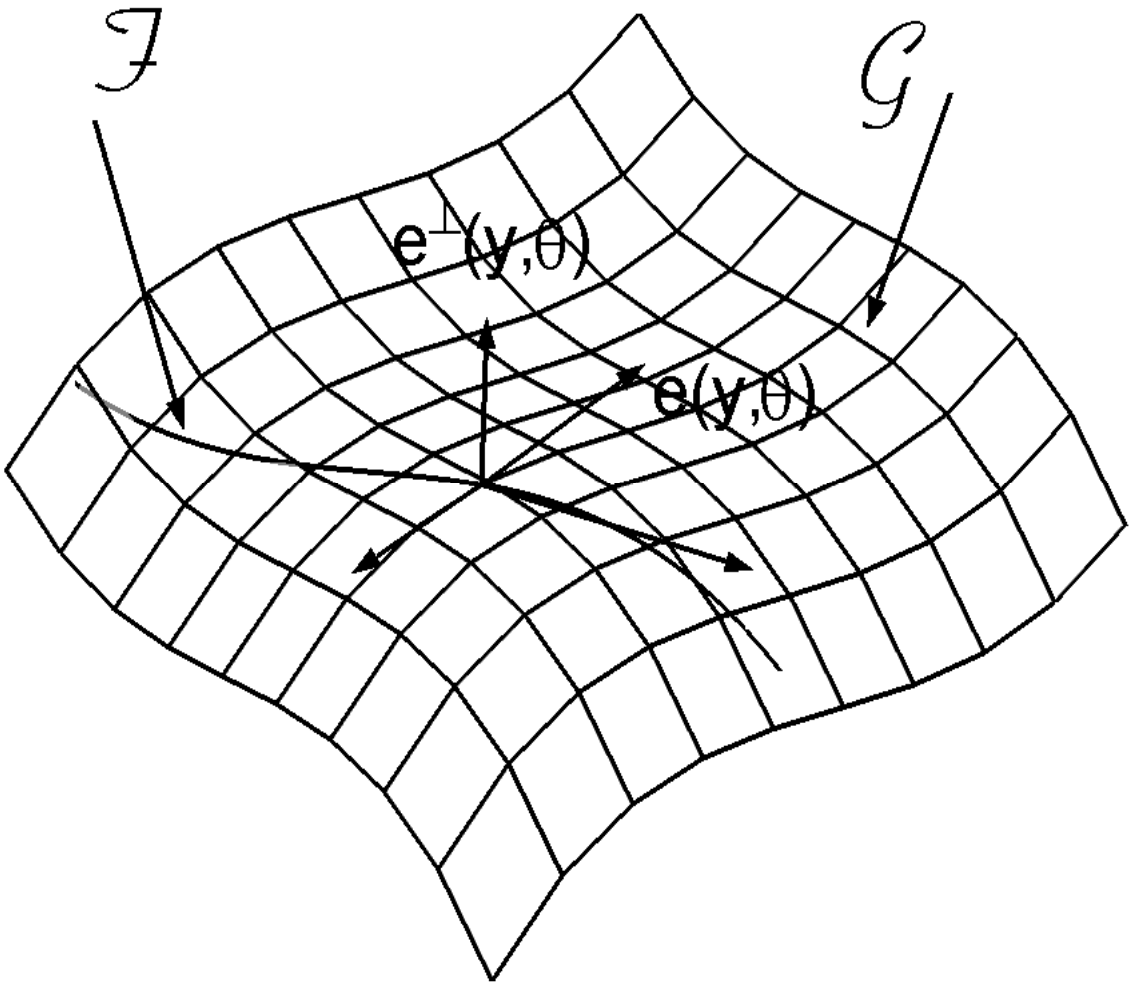

Figure 3.1: Geometrical interpretation of the construction of tests insensitive to Pitman drift.

simulations the following aspects were studied. First, the accuracy of setting the test threshold to $F^{-1}_{\chi^2_Q}(1-\alpha)$ for a level α test was evaluated. Second, we evaluated how fast the power of the test approaches 1, as the number of samples increases, and the accuracy of the finite sample power approximation (3.37). Finally, the sensitivity of the tests to a misspecified model is examined. The threshold adjustment procedure and the construction of tests that are orthogonal to deviations from the model are

demonstrated.

3.5.1 Direction Finding in Array Signal Processing

For a review of the problem of direction finding using antenna arrays see e.g. [41] or [67]. The characterization of the MLE under possible model mismatch has been recently addressed in [47] and [126].

Here we adopt the standard narrow band model of [111]. We consider the estimation of the directions of two uncorrelated narrow band Gaussian sources using a uniform linear array of $P = 4$ sensors with $\lambda/2$ spacing between elements (λ is the wavelength of wavefronts propagating across the array). The received signal model is given by

$$y_t = D(\theta)s_t + w_t$$

where $y_t \in \mathcal{C}^P$ is the noisy data vector at the array elements,

$$D(\theta) = [d(\theta_1) \quad d(\theta_2)]$$

where $[d(\theta)]_p = \exp\{jp\pi\cos(\theta)\}$, $p = 0, 1, 2, 3$ is the steering vector, s_t contains the two signal components, and w_t is a temporally and spatially complex white circular Gaussian noise. This signal model corresponds to the so called stochastic signal model in which the received signal at the array is distributed as a temporally white zero-mean complex circular Gaussian random vector with covariance matrix $C(\theta) = D(\theta)K_sD^H(\theta) + \sigma^2 I$, where, due to an uncorrelated sources assumption, $K_s = \text{diag}(\sigma_{s1}^2, \sigma_{s2}^2)$, σ_{s1}^2 and σ_{s2}^2 are the two source variances, and σ^2 is the noise variance. Hence, the density of y is given by

$$f(y, \theta) = \frac{1}{\pi^P \det(C(\theta))} \exp\left[-y^H C^{-1}(\theta) y\right]. \tag{3.47}$$

The variances σ^2, σ_1^2, and σ_2^2 are assumed known. The only unknowns are the sources directions, $\theta = [\theta_1, \theta_2]^T$. In the simulations the true unknown parameters were taken to be $\theta = [\pi/2, \pi/2 + 0.4]^T$ and the other known parameters were set to $\sigma_{s1}^2 = \sigma_{s2}^2 = 1$, and $\sigma^2 = 2$. In Fig 3.2, the log-likelihood surface calculated from 200 samples is shown and it is seen that it has two relative maxima.

Recall that the global maximum validation function of Biernacki's test is given by

$$\begin{aligned} e(y,\theta) &= \log f(y;\theta) - \int \log f(y;\theta) f(y;\theta) dy \\ &= -\log\left(\pi^P\right) - \log\left(\det\left(C(\theta)\right)\right) - y^H C^{-1}(\theta) y \\ &\quad + \log\left(\pi^P\right) + \log\left(\det\left(C(\theta)\right)\right) \\ &\quad + \int y^H C^{-1}(\theta) y f(y;\theta) dy \\ &= P - y^H C^{-1}(\theta) y. \end{aligned}$$

Hence

$$\begin{aligned} h_n(\widetilde{\theta}_n) &= \frac{1}{n}\sum_{t=1}^{n} e(y_t, \widetilde{\theta}_n) \\ &= P - \frac{1}{n}\sum_{t=1}^{n} y_t^H C^{-1}(\widetilde{\theta}_n) y_t \\ &= P - \mathrm{tr}\left(C^{-1}(\widetilde{\theta}_n)\widehat{C}\right) \end{aligned}$$

where

$$\widehat{C} = \frac{1}{n}\sum_{t=1}^{n} y_t y_t^H.$$

Under the null hypothesis and assuming the model is correctly specified, a closed

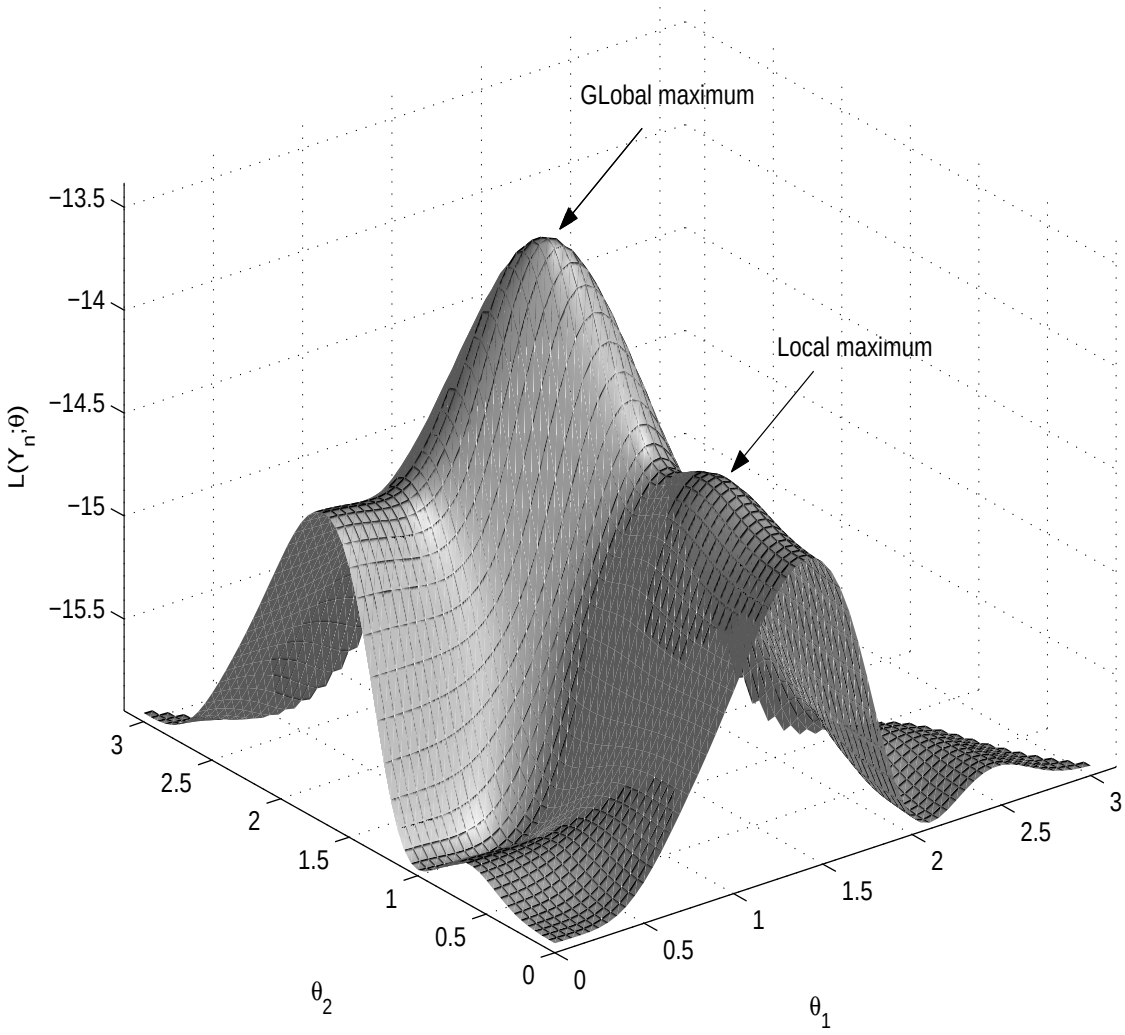

Figure 3.2: The log-likelihood function of the direction finding problem.

form expression for the variance can be computed through (3.30), where

$$\begin{aligned}\left[\overline{H}(\theta)\right]_{1,i} &= \int \partial e(y,\theta)/\partial\theta_i f(y,\theta)dy \\ &= \int y^H C^{-1}(\theta)\frac{\partial C(\theta)}{\partial\theta_i}C^{-1}(\theta)yf(y,\theta)dy \\ &= \mathrm{tr}\left(C^{-1}(\theta)\frac{\partial C(\theta)}{\partial\theta_i}\right), \quad i=1,2\end{aligned}$$

$$\begin{aligned}\int e^2(y,\theta)f(y,\theta)dy &= \int \left[P - y^H C^{-1}(\theta)y\right]^2 f(y,\theta)dy \\ &= P\end{aligned}$$

and $\widetilde{B}(\theta)$ is the FIM for this problem [61, p. 565], given by

$$\left[\overline{B}(\theta)\right]_{i,j} = \mathrm{tr}\left[C^{-1}(\theta)\frac{\partial C(\theta)}{\partial\theta_i}C^{-1}(\theta)\frac{\partial C(\theta)}{\partial\theta_j}\right]. \tag{3.48}$$

Hence

$$\overline{V}(\widetilde{\theta}_n) = P - \overline{H}(\widetilde{\theta}_n)\overline{B}(\widetilde{\theta}_n)\overline{H}^T(\widetilde{\theta}_n)$$

and the test statistic is given by

$$S_n = n\left[P - \mathrm{tr}\left(C^{-1}(\widetilde{\theta}_n)\widehat{C}\right)\right]^2 / \overline{V}(\widetilde{\theta}_n). \tag{3.49}$$

The threshold is set according to a χ^2 distribution with one degree of freedom.

We compare Biernacki's test to a test which is based on the real part of the first off-diagonal element of the covariance matrix. To compare the first off-diagonal element of the covariance matrix at the candidate relative maximum to its unconstrained estimate from the data, the global maximum validation function is taken to

be

$$e(y,\theta) = y^H My - \mathrm{tr}\left(MC(\theta)\right)$$

where M is the symmetric Toeplitz matrix whose first row is $[0,1,0,0]$, and hence

$$\begin{aligned} h_n(\widetilde{\theta}_n) &= \frac{1}{n}\sum_{t=1}^{n} e(y_t, \widetilde{\theta}_n) \\ &= \mathrm{tr}\left(M\widehat{C}\right) - \mathrm{tr}\left(MC(\widetilde{\theta}_n)\right). \end{aligned}$$

For this choice of $e(y,\theta)$ we have

$$\left[\overline{H}(\theta)\right]_{1,i} = -\mathrm{tr}\left(M\frac{\partial C(\theta)}{\partial \theta_i}\right), \quad i = 1,2 \tag{3.50}$$

and by [61, p. 564]

$$\begin{aligned} \int e^2(y,\theta) f(y,\theta) dy = & \\ & \int \left[y^H My - \mathrm{tr}\left(MC(\theta)\right)\right]^2 f(y,\theta) dy \\ & = \mathrm{tr}\left(MC(\theta)MC(\theta)\right). \end{aligned}$$

Hence

$$\overline{V}(\widetilde{\theta}_n) = \mathrm{tr}\left(MC(\widetilde{\theta}_n)MC(\widetilde{\theta}_n)\right) - \overline{H}(\widetilde{\theta}_n)\overline{B}(\widetilde{\theta}_n)\overline{H}^T(\widetilde{\theta}_n)$$

the test statistic is given by

$$S_n = n\left[\mathrm{tr}\left(M\widehat{C}\right) - \mathrm{tr}\left(MC(\widetilde{\theta}_n)\right)\right]^2 / \overline{V}(\widetilde{\theta}_n) \tag{3.51}$$

and, again, the threshold is set according to a χ^2 distribution with one degree of freedom.

The power performance of Biernacki's test and a Covariance based test were

evaluated for increasing n for levels that were set to 0.01 and 0.001. 1000 Monte Carlo iterations were used. At each iteration the global maximum and the local maximum were found and the tests were applied to both maxima to evaluate the performance. When the number of samples is very small (e.g. $n = 20$), the likelihood function my be distorted and the two relative maxima may collapse into one. Such cases were eliminated from the analysis. The results are summarized in Fig. 3.3. While not presented here, we observed that the empirical levels of both tests were in good agreement with the specified values.

Model Mismatch

In this section the performance of the tests (3.49) and (3.51) under model mismatch is evaluated. The assumed model used for the estimation is the same as in the previous section (3.47). The samples were generated according to the model (3.47) but with covariance matrix

$$C(\theta, \gamma) = D(\theta) K_s D^H(\theta) + \sigma^2 R(\gamma), \tag{3.52}$$

where $R(\gamma)$ is a symmetric Toeplitz matrix whose first row is $[1, \gamma, \gamma^2, \gamma^3]$, which corresponds to a first order AR spatial noise covariance [85], and in the simulation $\gamma = 0.1$.

For both Biernacki's test and the covariance based test the effect of model mismatch on the level was evaluated for three cases: (a) The increase in level due to model mismatch when the tests are performed without any adjustment, (b) The threshold correction described in Sec. 3.4.1, and (c) The performance of the orthogonal counterparts given in Sec. 3.4.2.

To perform the threshold correction described in Sec. 3.4.1, the Kullback-Leibler distance needs to be estimated. In the simulation, it was assumed that it is known

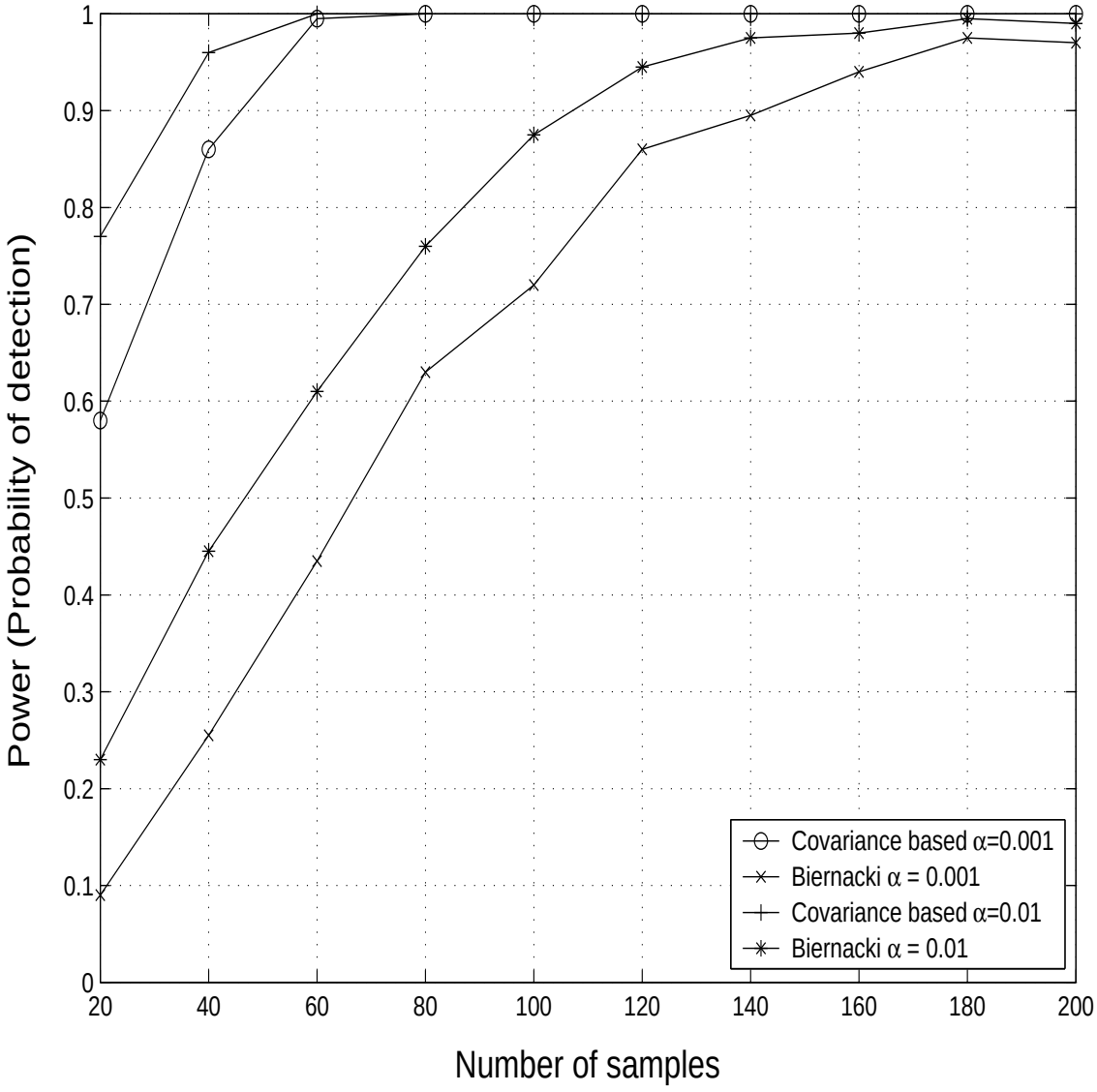

Figure 3.3: Direction finding: power when the model is correctly specified.

that the parameter γ, which controls the deviation from the model, ranges between zero (correct model) and 0.1. At every Monte Carlo iteration, given a relative maximum $\widetilde{\theta}_n$,

$$c = \max_{\gamma \in [0,1]} D_1 \left(\widetilde{f}(y; \widetilde{\theta}_n, \gamma) || f(y; \widetilde{\theta}_n) \right)$$

was computed, using the known formula for the Kullback-Leibler distance between two Gaussian densities (e.g. [74]), where $f(y;\theta)$ is given in (3.47) and $\widetilde{f}(y;\theta,\gamma)$ is the same density but with covariance matrix $C(\theta,\gamma)$ (3.52). Then, the null hypothesis was rejected if

$$S_n > F^{-1}_{\chi^2_Q(n[\exp(2c)-1])}(1-\alpha).$$

The simulation results show that, as anticipated, the level decreases rather than increases with the number of samples (see Fig. 3.4, where CT is a shorthand notation for 'corrected threshold').

To construct the orthogonal counterparts of the two tests, $e^{\perp}(y,\theta)$ is found through (3.44). For Biernacki's test the elements of $E(\beta)$ (3.45), which is a 1×3 vector is this case, are given by

$$[E(\beta)]_i = -\text{tr}\left(C^{-1}(\beta) \frac{\partial C^{-1}(\beta)}{\partial \beta_i} \right), \quad i = 1,2,3$$

where, as defined earlier, $\beta = [\theta^T, \gamma]^T$. For the covariance based test the elements of $E(\beta)$ are given by

$$[E(\beta)]_i = \text{tr}\left(M \frac{\partial C^{-1}(\beta)}{\partial \beta_i} \right), \quad i = 1,2,3.$$

The FIM $\widetilde{B}(\beta)$ is also available in closed form as given in (3.48). Using the closed forms for $E(\beta)$ and $\widetilde{B}(\beta)$, the variance for the two tests was computed through (3.46). In Fig. 3.4 it is seen that while the original tests suffer from increased level as the

number of samples increase, the orthogonal counterparts are unaffected by this type of model mismatch.

3.5.2 Estimation of Gaussian Mixture Parameters

The problem of estimation of Gaussian mixture parameters arises in both non-parametric density estimation (see e.g. [90] and references therein) and a variety of clustering problems (see e.g. [43] and references therein). The MLE for this problem is usually found by using the EM algorithm [81]. In [43], the authors describe a method that finds the global maximum with good performance. However, even this state of the art method is not certain to find the global maximum, and therefore, tests for global maximum are useful.

Here we consider the univariate case, in which the independent scalar measurements are generated by the following two component univariate Gaussian mixture density

$$f(y;\theta) = \sum_{l=1}^{2} \frac{p_l}{\sqrt{2\pi\sigma_l^2}} \exp\left\{-\frac{(y-\eta_l)^2}{2\sigma_l^2}\right\} \tag{3.53}$$

where the parameter vector consists of the two means $\theta = [\eta_1\ \eta_2]^T$. The number of components, the variances, and the mixing probabilities are assumed known. In the simulation, the true parameter is $\theta = [0,\ 3]^T$, the variances are $\sigma_1^2 = 1$ and $\sigma_1^2 = 0.5$, the mixing probabilities are $p_1 = 1 - p_2 = 0.35$ and it is known that $\Theta = [-1,\ 4] \times [-1,\ 4]$. The likelihood surface over Θ of a realization of 200 samples generated according to this model is presented in Fig. 3.5 and two relative maxima appear.

The performance of the global maximum tests was evaluated as the number of samples n increases. 1000 Monte Carlo iterations were generated. At each iteration, Biernacki's test and a mean based test were performed on both the global and the

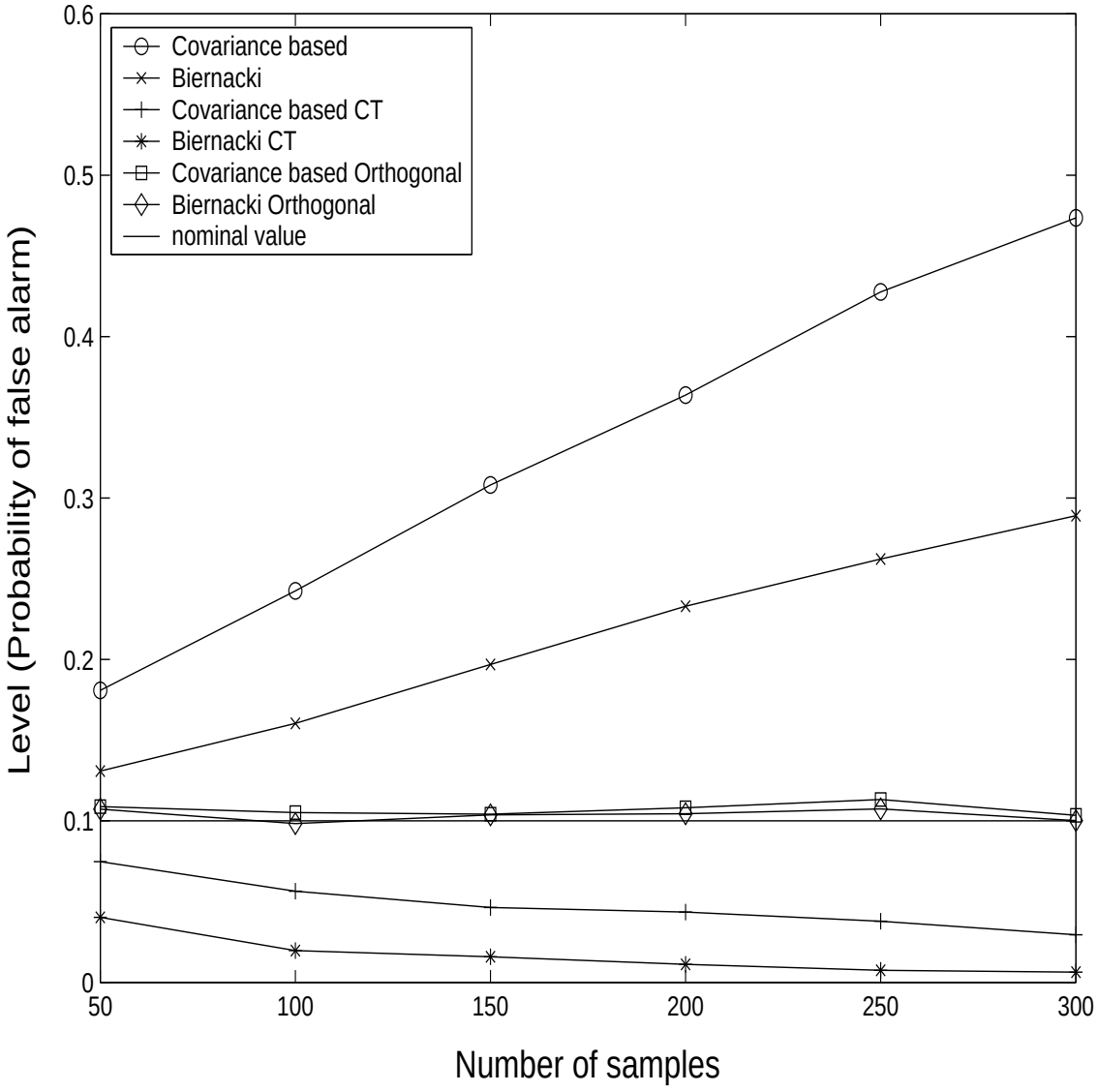

Figure 3.4: Direction finding: level under model mismatch.

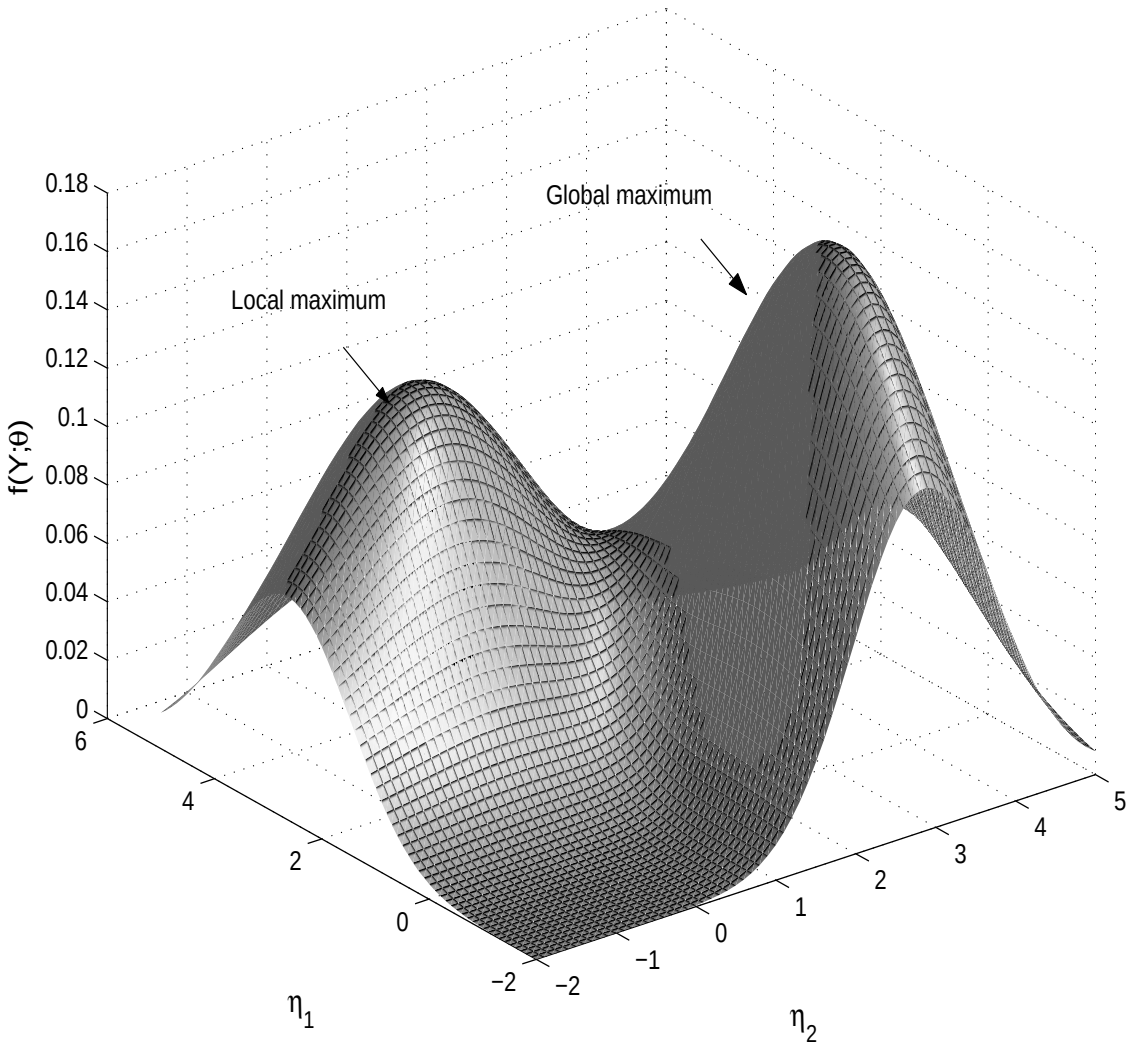

Figure 3.5: The likelihood function of the Gaussian mixture distribution.

local maxima. As in the previous section, Biernacki's global maximum validation function is given by

$$e(y, \theta) = \log f(y; \theta) - \int \log f(y; \theta) f(y; \theta) dy \tag{3.54}$$

and therefore,

$$h_n(\widetilde{\theta}_n) = \frac{1}{n}\sum_{t=1}^{n} \log f(y_t;\widetilde{\theta}_n) - \int \log f(y;\widetilde{\theta}_n) f(y;\widetilde{\theta}_n) dy.$$

A closed form expression to the integral in (3.54) is not available. Hence, in the simulations, numerical integration is used. The variance $V_n(\widetilde{\theta}_n)$ required for the construction of the test statistic S_n (3.19) was calculated through (3.13). Note that $H_n(\theta)$, required for calculating $V_n(\widetilde{\theta}_n)$, simplifies under the null hypothesis, i.e. $\widetilde{\theta}_n = \widehat{\theta}_n$, to

$$\begin{aligned} H_n(\widetilde{\theta}_n) &= \frac{1}{n}\sum_{t=1}^{n} \nabla_\theta^T e(y_t,\theta)\Big|_{\theta=\widetilde{\theta}_n} \\ &= \frac{1}{n}\sum_{t=1}^{n} \nabla_\theta^T \log f(y;\theta) \\ &\quad - \int \nabla_\theta^T \log f(y;\theta) f(y;\theta) dy \\ &\quad - \int \log f(y;\theta) \nabla_\theta^T f(y;\theta) dy\Big|_{\theta=\widetilde{\theta}_n} \\ &= -\int \log f(y;\theta) \nabla_\theta^T f(y;\theta) dy\Big|_{\theta=\widetilde{\theta}_n} \end{aligned}$$

which was calculated in the simulation by numerical integration.

The global maximum validation function of the mean based test is given by

$$e(y,\theta) = y - [p\eta_1 + (1-p)\eta_2]$$

which leads to

$$h_n(\widetilde{\theta}_n) = \frac{1}{n}\sum_{t=1}^{n} y_t - (p\widetilde{\eta}_1 + (1-p)\widetilde{\eta}_2). \tag{3.55}$$

Similar to the previous test, the variance required for the test statistic was calculated

through (3.13), where, for this test, the vector $H_n(\widetilde{\theta}_n)$ is given by

$$H_n(\widetilde{\theta}_n) = -\left[p,\ (1-p)\right].$$

The level of the tests was set to 0.01 and the empirical power was estimated from $10,000$ Monte Carlo iterations and compared to the analytic approximation (3.37). The results are summarized in Fig. 3.6 and it can be seen that the analytical power approximation predicts the empirical power well. It can be seen that the power of the mean based test is better than that of Biernacki's test. For other choices of parameters different results may be obtained. While not reported here, the empirical level of both tests was in good agreement with its specified value.

3.5.3 Estimation of Superimposed Exponentials in Noise

For a review of the problem of estimating the parameters of superimposed exponentials in noise see, e.g., [111]. Consider the following model

$$y_t = \sum_{k=1}^{K} \alpha_k \exp\{j\Omega_k t\} + w_t, \quad t = 1, \ldots, n$$

where w_t is a white circular Gaussian noise with unknown variance σ^2. The unknown parameters are the frequencies of the exponentials $[\Omega_1, \ldots, \Omega_K]$, their complex valued amplitudes $[\alpha_1, \ldots, \alpha_K]$ and the noise variance. The number K of components is assumed known and was set to 3, hence there are 10 unknown parameters. The unknown parameters were set to $[\Omega_1, \Omega_2, \Omega_3] = [0.4,\ 0.5,\ 0.6]$, $[\alpha_1, \alpha_2, \alpha_3] = [\exp(j2),\ 0.8\exp(j3),\ 1.2\exp(j5)]$, and $\sigma^2 = 1$.

Under this generating model, the data are independent but not identically distributed. They are distributed as non-zero time-varying mean circular Gaussian pro-

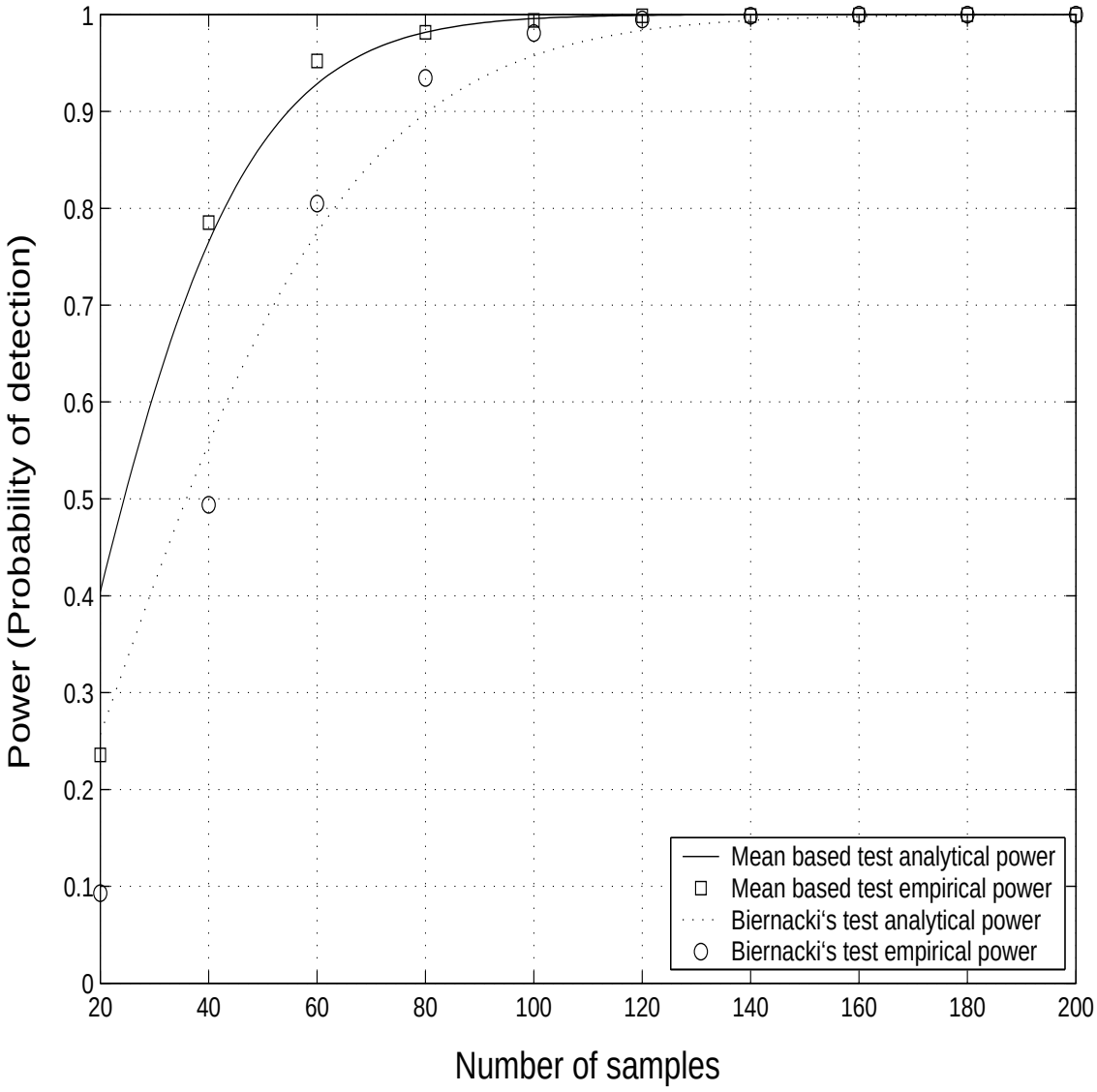

Figure 3.6: Gaussian mixture: empirical power vs. its analytic prediction, when the level is set to 0.01.

cess. Hence, the treatment in Sec. 3.2.1 does not cover this problem. Furthermore, since the MLE for this problem is super efficient [98], the more general framework of White [124] for constructing tests in dynamical models does not cover this problem either. However, a detailed statistical asymptotic analysis for this problem is available in the literature and can be used to construct a test for global maximum. In particular, in [98] it was shown that the MLE is asymptotically normal distributed under an appropriate normalization. Based on this analysis, we propose a test which is based on the autocorrelation function. In particular, our test is based on the fact that at the true parameter,

$$\begin{aligned}&\mathrm{E}\left\{\left[y_t-\sum_{k=1}^{K}\alpha_k\exp(j\Omega_k t)\right]\times\right.\\&\left.\left[y_{t-1}-\sum_{k=1}^{K}\alpha_k\exp(j\Omega_k(t-1))\right]^*\right\}=\\&\mathrm{E}\left\{e_t e_{t-1}^*\right\}=0,\end{aligned}$$

and hence, given the local maximum $\widetilde{\theta}_n$, we construct a test from the real part of the statistic

$$\begin{aligned}h_n(\widetilde{\theta}_n)=&\frac{1}{n-1}\sum_{t=2}^{n}\left[y_t-\sum_{k=1}^{K}\widetilde{\alpha}_k\exp(j\widetilde{\Omega}_k t)\right]\times\\&\left[y_{t-1}-\sum_{k=1}^{K}\widetilde{\alpha}_k\exp(j\widetilde{\Omega}_k(t-1))\right]^*.\end{aligned}$$

It is shown in the Appendix that under the null hypothesis, the real part of this statistic is asymptotically distributed as a zero-mean Gaussian random variable with variance $\sigma^2/2$. Hence, since under the null hypothesis $\widetilde{\sigma}^2$ is a consistent estimator

for σ^2, the statistic

$$n\frac{\left(\Re\{h_n(\widetilde{\theta}_n)\}\right)^2}{\widetilde{\sigma}^2/2}$$

is asymptotically χ^2 distributed with one degree of freedom, and can be used to discriminate between local and global maxima. In Fig. 3.7 the performance of this test is presented when the level is set to 0.01. The empirical level and power of the test were estimated from 1000 Monte Carlo iterations. It is seen that the asymptotic approximation to the level α is accurate for n greater then 300 and the power of the test approaches 1 when n is greater then 100.

3.6 Concluding Remarks

This paper has investigated a method for detecting a case in which a local search for the maximum likelihood has stagnated at a local maximum. This is a useful tool for exploring solutions of the global optimization problem associated with the ML method. Because existing tests are sensitive to model mismatch, the general treatment given here is necessary for practical implementation of this tool. The framework given for the construction of tests and the power analysis enable us to pose fundamental questions of optimality: Given a statistical model, what is the best choice of $e(y, \theta)$ in terms of achieving maximum power for a given level with minimum sensitivity to model mismatch? This remains an open question.

It is possible to generalize the above concept to non-i.i.d. measurements. A unified treatment of the MLE under a possible model mismatch and the construction of model mismatch tests for dynamic models is given in [124] and an example is which the measurements are i.n.i.d. was treated in Sec. 3.5.3. The concept of using a

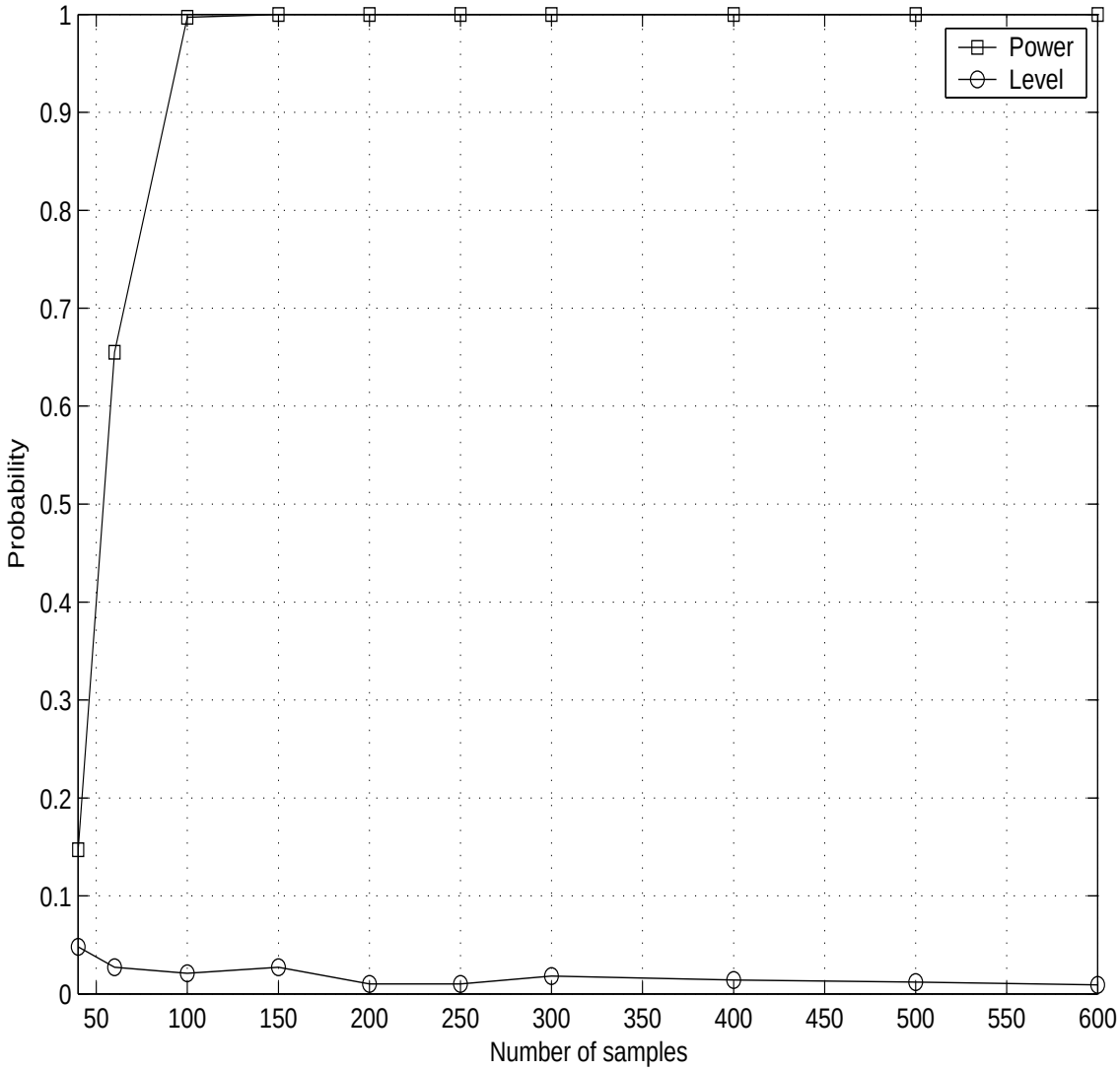

Figure 3.7: Exponentials in noise: performance when the model is correctly specified.

statistical test for discriminating between global and local maxima can be generalized to other M-estimators [55], or any other optimization problem in which a statistical characterization of the global maximum is available.

3.7 Asymptotic Distribution of M-tests

The proof follows White's methodology [124]. Given the assumptions that the elements of $e(y,\theta)$ are twice differentiable with respect to θ for every y, and that the elements of the vector $\nabla_\theta e(y,\theta)$ and the matrices $e(y,\theta)\nabla_\theta^T \log f(y,\theta)$ and $e(y,\theta)e^T(y,\theta)$ are dominated by functions integrable with respect to G for all $\theta \in \Theta$, the mean value theorem for random functions, given as Lemma 3 in [57], guarantees the existence of measurable Θ-valued functions $\overline{\theta}_n$ such that

$$\sqrt{n}h_n(\widehat{\theta}_n) = \sqrt{n}h_n(\theta^*) + H_n(\overline{\theta}_n)\sqrt{n}\left(\widehat{\theta}_n - \theta^*\right) \tag{3.7.56}$$

where each $\overline{\theta}_n$ lies on the segment joining $\widehat{\theta}_n$ and θ^*. Each row of H_n depends on a different $\overline{\theta}_n$, but since it makes no difference asymptotically, the above shorthand notation is used. From (3.5) $\sqrt{n}\left(\widehat{\theta}_n - \theta^*\right)$ converges in distribution. Furthermore, $\widehat{\theta}_n \overset{a.s.}{\to} \theta^*$ and therefore $\overline{\theta}_n \overset{a.s.}{\to} \theta^*$ as well. From Theorem 2 in [57], applied on the elements of $H_n(\theta)$, we have $H_n(\theta) \overset{a.s.}{\to} H(\theta)$ uniformly in θ, and therefore using Lemma 3.1 of White [121], $H_n(\overline{\theta}_n) - H(\theta^*) \overset{a.s.}{\to} 0$. Using these intermediate results we obtain from $2c.4(xa)$ of Rao [97] that

$$\left[H_n(\overline{\theta}_n) - H(\theta^*)\right]\sqrt{n}\left(\widehat{\theta}_n - \theta^*\right) \overset{P}{\to} 0. \tag{3.7.57}$$

Equation (A.2) of [122] asserts that

$$A^{-1}(\theta^*)\frac{1}{\sqrt{n}}\sum_{t=1}^{n}\nabla \log f(y_t,\theta^*) + \sqrt{n}\left(\widehat{\theta}_n - \theta^*\right) \overset{P}{\to} 0.$$

Therefore, by the finiteness of $H(\theta^*)$, we have

$$H(\theta^*) \times$$
$$\left[A^{-1}(\theta^*)\frac{1}{\sqrt{n}}\sum_{t=1}^{n}\nabla \log f(y_t,\theta^*) + \sqrt{n}\left(\widehat{\theta}_n - \theta^*\right)\right]$$
$$\xrightarrow{P} 0.$$

Adding and subtracting $H_n(\overline{\theta}_n)\sqrt{n}\left(\widehat{\theta}_n - \theta^*\right)$ and rearranging terms, we obtain

$$\left[H(\theta^*) - H_n(\overline{\theta}_n)\right]\sqrt{n}\left(\widehat{\theta}_n - \theta^*\right) +$$
$$H_n(\overline{\theta}_n)\sqrt{n}\left(\widehat{\theta}_n - \theta^*\right) +$$
$$H(\theta^*)A^{-1}(\theta^*)\frac{1}{\sqrt{n}}\sum_{t=1}^{n}\nabla \log f(y_t,\theta^*) \xrightarrow{P} 0.$$

But from (3.7.57) the first term converges to zero in probability, and hence,

$$H_n(\overline{\theta}_n)\sqrt{n}\left(\widehat{\theta}_n - \theta^*\right) +$$
$$H(\theta^*)A^{-1}(\theta^*)\frac{1}{\sqrt{n}}\sum_{t=1}^{n}\nabla \log f(y_t,\theta^*) \xrightarrow{P} 0.$$

Substituting $H_n(\overline{\theta}_n)\sqrt{n}\left(\widehat{\theta}_n - \theta^*\right) = \sqrt{n}h_n(\widehat{\theta}_n) - \sqrt{n}h_n(\theta^*)$ from (3.7.56), adding and subtracting $\sqrt{n}h(\theta^*)$, and rearranging terms, we obtain

$$\sqrt{n}\left[h_n(\widehat{\theta}_n) - h(\theta^*)\right] - \frac{1}{\sqrt{n}}\sum_{t=1}^{n}[e(y_t,\theta^*) - h(\theta^*) -$$
$$H(\theta^*)A^{-1}(\theta^*)\nabla \log f(y_t,\theta^*)\big] \xrightarrow{P} 0.$$

From the Lindeberg-Lévy central limit theorem the second term converges in probability to a zero mean multivariate normal density, with covariance matrix $V(\theta^*)$ and therefore, from $2c.4(xd)$ of Rao [97], so does the first term, and the first part of the

theorem is proved. The consistency of $V_n(\widehat{\theta}_n)$ for $V(\theta^*)$ follows from Lemma 3.1 of White [121] given the assumptions, and the consistency guarantees that $V_n^{-1}(\widehat{\theta}_n)$ exists for sufficiently large n, since the determinant of a matrix is a continuous function of its elements. The last part of the theorem follows from Lemma 3.3 of White [120] and the proof is completed.

3.8 Asymptotic Distribution of the Test Statistic for Exponentials in Noise

The derivation is given under the null hypothesis, hence $\widetilde{\theta}_n = \widehat{\theta}_n$. Using the mean value theorem we obtain

$$h_n(\widehat{\theta}_n) = h_n(\theta^0) + \nabla^T h_n(\overline{\theta})(\widehat{\theta}_n - \theta^0) \quad a.s..$$

Using the martingale central limit theorem [22] with the filtration

$$\{\mathcal{F}_t = \sigma(e_1, \ldots, e_t)\},$$

we obtain that $h_n(\theta^0)$ converges in distribution to a zero-mean Gaussian random variable with variance $\sigma^2/2$, since $\mathrm{E}\{h_n(\theta^0)|\mathcal{F}_{n-1}\} = 0$ and $\mathrm{E}\{h_n^2(\theta^0)\} = \sigma^2/2$. Next, we show that the second term is $\mathrm{o}_P(1)$. First split the second term into two components

$$\begin{aligned}\nabla^T h_n(\overline{\theta})(\widehat{\theta}_n - \theta^0) = & n^{-3/2}\nabla_\Omega^T h_n(\overline{\theta}) n^{3/2}(\widehat{\boldsymbol{\Omega}}_n - \boldsymbol{\Omega}^0) + \\ & n^{-1/2}\nabla_\alpha^T h_n(\overline{\theta}) n^{1/2}(\widehat{\boldsymbol{\alpha}}_n - \boldsymbol{\alpha}^0).\end{aligned}$$

It is possible to show that both $n^{-3/2}\nabla_{\Omega}^{T}h_n(\overline{\theta})$ and $n^{-1/2}\nabla_{\alpha}^{T}h_n(\overline{\theta})$ converge to zero in probability. Therefore, since it was shown in [98] that both $n^{3/2}(\widehat{\boldsymbol{\Omega}}_n - \boldsymbol{\Omega}^0)$ and $n^{1/2}(\widehat{\boldsymbol{\alpha}}_n - \boldsymbol{\alpha}^0)$ converge in distribution, we have that this term converges to zero in probability. This establishes the asymptotic normality of $h_n(\widehat{\theta}_n)$. In [98] it was also shown that $\widehat{\sigma}^2$ converges to the true value of σ^2 a.s.. Therefore, by Lemma 3.3 of White [120], we obtain that the test statistic is asymptotically χ^2 distributed.

CHAPTER 4

Classification Reduction of Policy Search

4.1 Introduction

There has been increased interest in applying tools from supervised learning to problems in reinforcement learning. The goal is to leverage techniques and theoretical results from supervised learning for solving the more complex problem of reinforcement learning [9]. In [70] and [42], classification is incorporated into approximate policy iterations. In [7], regression and classification are used to perform dynamic programming. In [84] nonlinear regression is coupled with Q-learning [112] to construct an approximate dynamic programming algorithm, and regression-type generalization errors are derived for the resulting estimated policy. Bounds on the performance of a policy that is built from a sequence of classifiers are derived in [72] and [73].

A common theme of these methods is viewing the multi-stage decision process as a sequence of single-stage decision processes and applying techniques from supervised learning to handle the computational complexity of the solution. This approach is reminiscent of Bellman's celebrated dynamic programming method [11] for finding the optimal policy for controlling a Markov decision process. The challenge in ap-

plying this approach to the RL problem is in controlling three sources of error: (1) without knowing the optimal policy, one cannot sample from the distribution that it induces on the stochastic system's state space, as a result is it difficult to determine how to allocate the approximation resources over the state space, (2) finding the optimal decision rule at a certain stage hinges on knowing the optimal decision rule for future stages, this is never available when approximate solutions are involved, and (3) without a model, ensemble expectations must be replaced by empirical averages, which leads to estimation errors. For example, in [7] the first source of error is handled by assuming that it is possible to sample from a distribution that is at least close to the one induced by taking optimal actions. In [84], the fact that the distribution induced by random action selection supports the one induced by the optimal policy, is used to bound the effect of model mismatch. The second source of error can be controlled by investigating the effect of using an approximate policy for future stages [7] [84]. Another approach is to bound from above the return from the optimal policy by the return from hindsight selection rules that maximizes the sum of reward on every trajectory and not just on the average [72], [125]. Finite sample upper bounds were used to bound the third source of errors in, e.g., [62], and [84].

Similar to [72], we adopt the generative model assumption of [62] and tackle the problem of estimating competitive policies for controlling a T-step stochastic decision process, within an infinite class of policies, from a set of trajectory trees of the decision process. Under the generative model assumption, it is possible to generate realizations of the system's evolution for arbitrary histories. In [72] the T-step reinforcement learning problem was converted to a set of weighted classification problems by trying to fit a set of classifiers, one per each decision epoch, to the collection of the maximal reward paths on the trajectory trees. A bound relates the performance of these classifiers, for the task of fitting the maximal paths, to

the performance of the policy constructed by combining the single stage classifiers. When the process is stochastic, the actions that maximize the instantaneous sum of rewards are often not the actions that maximize the expected sum of reward. This mismatch leads to an inherent error that does not approach zero as the number of trajectory trees grows.

In this paper we take a different approach. Through an approximate dynamic programming algorithm we estimate a competitive policy by solving a sequence of single-stage reinforcement learning subproblems that are further reduced to supervised learning problems. Our single-stage reduction is exact and is different from the one proposed in [72]; it gives more weight to regions of the state space in which the difference between the possible actions in terms of future reward is large, rather than giving more weight to regions in which the maximal future reward is large. Since minimizing the empirical $0-1$ loss associated with the supervised learning problems is often intractable, a common strategy of many off-the-self methods is to instead minimize a surrogate loss function. Using a recent result from the classification literature [8], we analyze the effect of this type of surrogate approximation on our estimated policy.

Finally, we derive finite sample generalization error bounds of the type given in [84] for the policy estimated by the proposed algorithm. The approach we take is similar to the one in [84]. Namely, we first write the generalization error in terms of measures whose empirical counterparts are minimized by the algorithm, and then invoke uniform convergence results to bound these terms. However, the rates we establish are faster than the ones in [84], except for the case in which the approximation class is a linear space, for which we establish the same rates. Our finite sample bounds are also closely related to the convergence rate analysis in [62]. While in [62] it is assumed that all decision rules are optimized simultaneously, which poses an

intractable optimization problem, it is shown here that the same convergence rates apply for the case in which the decision rules are estimated sequentially and the estimated decision rules for later stages define the optimization problem for earlier stages.

4.2 Preliminaries

We consider a discrete finite horizon stochastic (but not necessarily Markovian) decision process and adopt the notation in [84]. In a stochastic decision process an agent collects observations of a system's state and takes actions which effect the system's future states. Each observation is, in general, a composition of state variables and noisy measurements of partially observable state variables. The actions belong to a finite set of actions called the action space $\mathcal{A} = \{0, 1, \ldots, L\}$. We assume for simplicity and without loss of generality, that $\mathcal{A}$ is the same for all decision epochs. A trajectory though the decision processes is the sequence $O_0, A_0, O_1, A_1, \ldots, O_T, A_T, O_{T+1}$ of random variables, where $O_t \in \mathcal{O}_t$ and $A_t \in \mathcal{A}$ are the observation and action at time t, $t = 0, 1, \ldots, T$, and $O_{T+1} \in \mathcal{O}_{T+1}$ is the final observation after which the agent does not take an action. Uppercase letters denote random variables and lowercase letters denote their realizations. Denote $(O_0, O_1, \ldots, O_t)$ and $(A_0, A_1, \ldots, A_t)$ by $\mathbf{O}_t$ and $\mathbf{A}_t$, respectively. The decision process begins as the agent obtains its first observation O_0. At every time t, $t = 0, 1, \ldots, T$, the agent takes an action A_t, observes O_{t+1} and receives a reward $r_t(\mathbf{O}_t, \mathbf{A}_t, O_{t+1})$ and we assume that the reward takes values in $[0, 1]$. The goal of the agent is to select its action so that the expected sum of rewards is maximized. A non-stationary policy is a sequence of action selection rules for each of the decision epochs and is denoted by $\pi = (\pi_0, \pi_1, \ldots, \pi_T)$. A deterministic selection rule π_t maps possible histories $(\mathbf{o}_t, \mathbf{a}_{t-1})$ to the action space.

When a deterministic policy is used to choose actions, the joint density function of the sequence of random variables $O_0, A_0, O_1, A_1, \ldots, O_T, A_T, O_{T+1}$ is given by

$$f_0(o_0)I(a_0 = \pi_0(o_0)) \prod_{t=1}^{T} f_t(o_t|\mathbf{o}_{t-1}, \mathbf{a}_{t-1})I(a_t = \pi_t(\mathbf{o}_t, \mathbf{a}_{t-1}))f_T(o_{T+1}|\mathbf{o}_T, \mathbf{a}_T), \quad (4.2.1)$$

where I is the indicator function, which equals one when its argument is true and zero otherwise, $f_0(o_0)$ is the density of the initial observation, and $f_t(o_t|\mathbf{o}_{t-1}, \mathbf{a}_{t-1})$ is the density of O_t given the past observations and actions. A random selection rule π_{p_t} specifies a conditional distribution over the action space conditioned on past observations and actions, $p_t(\cdot|\mathbf{o}_t, \mathbf{a}_{t-1})$, according which the actions are selected at random. When a random policy is used to choose actions, the joint density function of the sequence $O_0, A_0, O_1, A_1, \ldots, O_T, A_T, O_{T+1}$ is instead given by

$$f_0(o_0)p_0(a_0|o_0) \prod_{t=1}^{T} f_t(o_t|\mathbf{o}_{t-1}, \mathbf{a}_{t-1})p_t(a_t|\mathbf{o}_t, \mathbf{a}_{t-1})f_T(o_{T+1}|\mathbf{o}_T, \mathbf{a}_T), \quad (4.2.2)$$

where p_t is the conditional action distribution associated with the random decision rule π_{p_t}. In general a policy can be a mix of random and deterministic decision rules. We denote an expectation of a function of $\mathbf{O}_{T+1}, \mathbf{A}_T$ under policy π by

$$\mathrm{E}_\pi \left\{ f(\mathbf{O}_{T+1}, \mathbf{A}_T) \right\}.$$

Note that the subscript π specifies the density with which we integrate in the expectation. For example, the expectation of $f(\mathbf{O}_{T+1}, \mathbf{A}_T)$ under a policy which dictates random action selection for times $t = 0, 1, \ldots, j$ according to the conditional distributions $p_0, p_1, \ldots, p_j$, and deterministic action selection for time $j+1, j+2, \ldots, T$

according to the mappings $\pi_{j+1}, \pi_{j+2}, \ldots, \pi_T$ is denoted by

$$\mathrm{E}_{\pi_{p_0}, \pi_{p_1}, \ldots, \pi_{p_j}, \pi_{j+1}, \pi_{j+2}, \ldots, \pi_T} \left\{ f(\mathbf{O}_{T+1}, \mathbf{A}_T) \right\}.$$

It is seen from (4.2.1) and (4.2.2) that when computing the expectation of a function $f(\mathbf{O}_{j+1}, \mathbf{A}_j)$, which does not depend on actions and observations beyond time $t = j + 1$, the decision rules for actions $A_{j+1}, \ldots, A_T$ can be specified arbitrarily, and hence will sometimes be denoted by

$$\mathrm{E}_{(\pi_0, \pi_1, \ldots, \pi_j, \cdot)} \left\{ f(\mathbf{O}_{j+1}, \mathbf{A}_j) \right\}.$$

Alternatively, when computing expectations of functions conditioned on $\mathbf{O}_t = \mathbf{o}_t, \mathbf{A}_t = \mathbf{a}_t$, one can specify arbitrary action selection rules up to time t, and hence the expectation of a function $f(\mathbf{O}_{T+1}, \mathbf{A}_T)$ conditioned on $\mathbf{O}_t = \mathbf{o}_t, \mathbf{A}_t = \mathbf{a}_t$ will sometimes be denoted by

$$\mathrm{E}_{(\cdot, \pi_{t+1}, \pi_{t+2}, \ldots, \pi_T)} \left\{ f(\mathbf{O}_{T+1}, \mathbf{A}_T) | \mathbf{O}_t = \mathbf{o}_t, \mathbf{A}_t = \mathbf{a}_t \right\}.$$

The value function of a policy π for observation o_0 is the expected sum of rewards conditioned on the value of the initial observation, when actions are taken according to the policy and it is denoted by

$$V_\pi(o_0) = \mathrm{E}_\pi \left\{ \sum_{t=0}^{T} r(\mathbf{O}_t, \mathbf{A}_t, O_{t+1}) | O_0 = o_0 \right\}. \tag{4.2.3}$$

The t-value function of a policy π is the expected sum of rewards under that policy

from time t on, conditioned on the observations $\mathbf{O}_t$ and actions $\mathbf{A}_{t-1}$:

$$V_{\pi,t}(\mathbf{o}_t, \mathbf{a}_{t-1}) = \mathrm{E}_\pi \left\{ \sum_{j=t}^{T} r(\mathbf{O}_j, \mathbf{A}_j, O_{j+1}) | \mathbf{O}_t = \mathbf{o}_t, \mathbf{A}_{t-1} = \mathbf{a}_{t-1} \right\}. \tag{4.2.4}$$

Note that $V_{\pi,0}(o_0) = V_\pi(o_0)$. It is seen from (4.2.1) and (4.2.2) that due to the conditioning on $\mathbf{O}_t = \mathbf{o}_t, \mathbf{A}_{t-1} = \mathbf{a}_{t-1}$, only the last $T - t + 1$ decision rules of π define the underlying distribution that enters into the expectation in (4.2.4). Hence, $V_{\pi,t}(\mathbf{o}_t, \mathbf{a}_{t-1})$ will sometimes be denoted by $V_{(\cdot,\pi_t,\pi_{t+1},\ldots,\pi_T),t}(\mathbf{o}_t, \mathbf{a}_{t-1})$

The optimal policy $\pi^* = (\pi_0^*, \pi_1^*, \ldots, \pi_T^*)$ is the policy that maximizes the value function

$$V_{\pi^*}(o_o) = \max_\pi V_\pi(o_0) \tag{4.2.5}$$

simultaneously for all $o_0 \in \mathcal{O}_0$. It is well known that the optimal policy satisfies

$$V_{\pi^*,t}(\mathbf{o}_t, \mathbf{a}_{t-1}) = \max_\pi V_{\pi,t}(\mathbf{o}_t, \mathbf{a}_{t-1}),$$

for every t and can be found through dynamic programming [11], [93]: starting from $V_{\pi^*,T+1} = 0$, solve for $t = T, T-1, \ldots, 0$

$$V_{\pi^*,t}(\mathbf{o}_t, \mathbf{a}_{t-1}) = \max_{a_t \in \mathcal{A}} \mathrm{E}\left\{ r(\mathbf{O}_t, \mathbf{A}_t, O_{t+1}) + V_{\pi^*,t+1}(\mathbf{O}_{t+1}, \mathbf{A}_t) | \mathbf{O}_t = \mathbf{o}_t, \mathbf{A}_t = \mathbf{a}_t \right\},$$
$$\pi_t^*(\mathbf{o}_t, \mathbf{a}_{t-1}) \in \arg\max_{a_t \in \mathcal{A}} \mathrm{E}\left\{ r(\mathbf{O}_t, \mathbf{A}_t, O_{t+1}) + V_{\pi^*,t+1}(\mathbf{O}_{t+1}, \mathbf{A}_t) | \mathbf{O}_t = \mathbf{o}_t, \mathbf{A}_t = \mathbf{a}_t \right\},$$

where, for time t, the expectation is computed with respect to the distribution of O_{t+1} conditioned on $(\mathbf{O}_t = \mathbf{o}_t, \mathbf{A}_t = \mathbf{a}_t)$, and hence, no policy needs to be specified.

Since π^* maximizes $V_\pi(o_0)$ for every o_0, it also maximizes the average value function

$$V(\pi) = \mathrm{E}\left\{ V_\pi(O_0) \right\} \tag{4.2.6}$$

where the expectation is taken with respect to the density $f_0(o_0)$ (4.2.1). It is the unique maximizer if the distribution of $f_0(o_0)$ is positive for all o_0. In this case, when optimizing over all possible policies, the maximization of (4.2.5) and (4.2.6) are equivalent. When optimizing (4.2.6) over a restricted class of policies, which does not contain the optimal policy, the distribution over the initial state specifies the importance of different regions of the observation space in terms of the approximation error. For example, assigning high probability to a certain region of the observation space will favor policies that well approximate the optimal policy over that region. Alternatively, maximizing (4.2.6) when the distribution of O_0 is a point mass at o_0 is equivalent to maximizing (4.2.5) for that specific o_0.

In the following sections, an algorithm is proposed, for fining a competitive policy from within a restricted class of policies that may or may not contain the optimal policy, from a limited amount of data. The algorithm uses a random policy to generate the distribution of the observations and, starting from the last stage, sequentially estimates the best decision rules for each of the stages given the decision rules that have already been obtained for the following stages. Here, we describe the procedure in terms of ensemble expectations. This can be seen as characterizing the behavior in the limit of an infinite data set. Consider a restricted class of policies of the form $\Pi = \{\pi = (\pi_0, \pi_1, \ldots, \pi_T) | \pi_0 \in \Pi_0, \pi_1 \in \Pi_1, \ldots, \pi_T \in \Pi_T\}$, that is, the policy class is a composition of $T+1$ classes of single stage decision rules. For every t, the class Π_t is a collection of decision rules that specify the action to take given any possible history $(\mathbf{o}_t, \mathbf{a}_{t-1})$. Define the policy $(\widehat{\pi}_0, \widehat{\pi}_1, \ldots, \widehat{\pi}_T) \in \Pi$ recursively as follows,

$$\widehat{\pi}_T \in \arg\max_{\pi_T \in \Pi_T} \mathrm{E}_{\pi_{q_0}, \pi_{q_1}, \ldots, \pi_{q_{T-1}}, \pi_T} \left\{ r(\mathbf{O}_T, \mathbf{A}_T, O_{T+1}) \right\}, \tag{4.2.7}$$

where $q_t(a|\mathbf{o}_t, \mathbf{a}_{t-1}) = 1/(L+1)\,\forall a \in \mathcal{A}$, and

$$\widehat{\pi}_t \in \arg\max_{\pi_t \in \Pi_t} \mathrm{E}_{\pi_{q_0}, \pi_{q_1}, \ldots, \pi_{q_{t-1}}, \pi_t, \widehat{\pi}_{t+1}, \ldots, \widehat{\pi}_T} \left\{ \sum_{j=t}^{T} r(\mathbf{O}_j, \mathbf{A}_j, O_{j+1}) \right\}, \tag{4.2.8}$$

for $t = T-1, T-2, \ldots, 0$. Note that

$$\begin{aligned}
&\mathrm{E}_{\pi_{q_0}, \pi_{q_1}, \ldots, \pi_{q_{t-1}}, \pi_t, \widehat{\pi}_{t+1}, \ldots, \widehat{\pi}_T} \left\{ \sum_{j=t}^{T} r(\mathbf{O}_j, \mathbf{A}_j, O_{j+1}) \right\} = \\
&\mathrm{E}_{\pi_{q_0}, \pi_{q_1}, \ldots, \pi_{q_{t-1}}, \pi_t, \widehat{\pi}_{t+1}, \ldots, \widehat{\pi}_T} \Bigg\{ r(\mathbf{O}_t, \mathbf{A}_t, O_{t+1}) + \\
&\quad \mathrm{E}_{(\cdot, \widehat{\pi}_{t+1}, \ldots, \widehat{\pi}_T)} \left\{ \sum_{j=t+1}^{T} r(\mathbf{O}_j, \mathbf{A}_j, O_{j+1}) | \mathbf{O}_{t+1}, \mathbf{A}_t \right\} \Bigg\} = \\
&\mathrm{E}_{\pi_{q_0}, \pi_{q_1}, \ldots, \pi_{q_{t-1}}, \pi_t, \widehat{\pi}_{t+1}, \ldots, \widehat{\pi}_T} \left\{ r(\mathbf{O}_t, \mathbf{A}_t, O_{t+1}) + V_{(\cdot, \widehat{\pi}_{t+1}, \ldots, \widehat{\pi}_T)}(\mathbf{O}_{t+1}, \mathbf{A}_t) \right\}
\end{aligned}$$

Hence, (4.2.8) is equivalent to

$$\widehat{\pi}_t \in \arg\max_{\pi_t \in \Pi_t} \mathrm{E}_{\pi_{q_0}, \pi_{q_1}, \ldots, \pi_{q_{t-1}}, \pi_t, \widehat{\pi}_{t+1}, \ldots, \widehat{\pi}_T} \left\{ r(\mathbf{O}_t, \mathbf{A}_t, O_{t+1}) + V_{(\cdot, \widehat{\pi}_{t+1}, \ldots, \widehat{\pi}_T), t}(\mathbf{O}_{t+1}, \mathbf{A}_t) \right\}.$$

The random action selection rules $q_t, t = 0, \ldots, T$ assign a positive density to any possible observations and actions history. Hence, if the policy class Π contains π^*, then $\widehat{\pi} = (\widehat{\pi}_0, \widehat{\pi}_1, \ldots, \widehat{\pi}_T) = \pi^*$.

4.3 The Data Generating Process

Following the generative model assumption of [62], we assume that the initial distribution of O_0 and the conditional distribution of O_{t+1} given $\mathbf{o}_t, \mathbf{a}_t$ are unknown but it is possible to generate realization of O_0 and realizations of O_{t+1} conditioned on arbitrary histories $\mathbf{o}_t, \mathbf{a}_t$. Furthermore, it is shown in [62] that a finite horizon stochas-

tic decision process with a generative model can be reduced to an equivalent process with a binary action space. Hence we assume hereafter that $\mathcal{A} = \{-1, 1\}$. Given the generative model, n trajectory trees are constructed in the following manner. The root of each tree is a realization of O_0. Given the realization of O_0, realizations of the next observation O_1 given the two possible actions, denoted by O_1^a, $a \in \mathcal{A}$, are generated. Note that in order to avoid notational explosion, this notation omits the dependence on the value of the initial observation O_0, but the likelihood of observing o_1^a at the leaf following a root whose value is o_0 is

$$f_1(o_1^a|o_0, a), \quad a = \pm 1.$$

The two realizations of O_1 are the roots of two subtrees. These iterations continue to generate a depth $T + 1$ tree. Denote by $O_t^{\mathbf{a}_{t-1}}$, where $\mathbf{a}_t = (a_0, a_1, \ldots, a_t)$, the random variable generated at the node that follows the sequence of actions $a_0, a_1, \ldots, a_{t-1}$. This random variable is a realization of O_t conditioned on the sequence of actions and sequence of observations that appear on the path of the tree that leads to it. Hence the likelihood of $O_t^{\mathbf{a}_{t-1}}$ taking value $o_t^{\mathbf{a}_{t-1}}$ is

$$f_t(o_t^{\mathbf{a}_{t-1}}|\mathbf{o}_{t-1}^{\mathbf{a}_{t-1}}, \mathbf{a}_{t-1}), \quad \mathbf{a}_{t-1} \in \{-1, 1\}^t,$$

where $\mathbf{o}_{t-1}^{\mathbf{a}_{t-1}}$ is a shorthand notation for $(o_0, o_1^{a_0}, o_2^{a_0,a_1}, \ldots, o_t^{\mathbf{a}_{t-1}})$, i.e., the realizations of the observations up to time $t-1$ on the path that leads to $O_t^{\mathbf{a}_{t-1}}$. In Fig. 4.1 a binary trajectory tree of depth $T+1 = 3$ is given. Let G be the collection of random variables that appear on the trajectory tree and denote by $\mathbf{E}$ the expectation with respect to these random variables.

Since we consider the problem of estimating good policies from n trajectory trees, it will be useful to express the average value function of a policy in terms of the

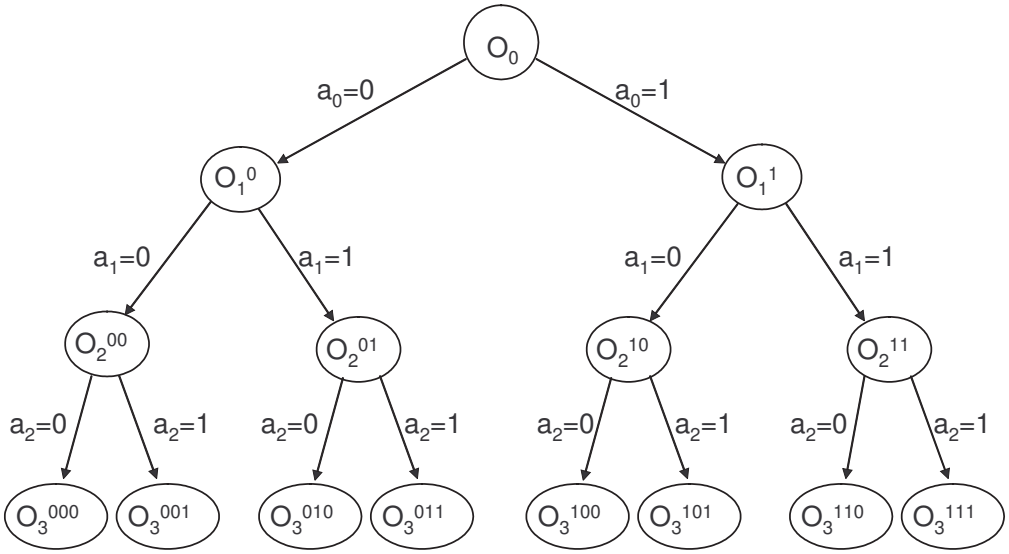

Figure 4.1: A binary trajectory tree of depth $T + 1 = 3$.

random variables that appear on the trajectory tree. For a deterministic policy π we have

$$\mathrm{E}_{\pi}\left\{\sum_{t=0}^{T} r\left(\mathbf{O}_t, \mathbf{A}_t, O_{t+1}\right)\right\} =$$
$$\mathrm{E}\left\{\sum_{\mathbf{a}_T \in \{-1,1\}^{T+1}} \prod_{t=0}^{T} I\left(\pi_t(\mathbf{O}_t^{\mathbf{a}_{t-1}}, \mathbf{a}^{t-1}) = a_t\right) \sum_{t=0}^{T} r(\mathbf{O}_t^{\mathbf{a}_{t-1}}, \mathbf{a}_t, O_{t+1}^{\mathbf{a}_t})\right\}.$$

To write the average value function of the policy $(\pi_{q_0}, \ldots, \pi_{q_{j-1}}, \pi_j, \ldots, \pi_T)$ we define the binary action variables $B_0, B_1, \ldots, B_{j-1}$ independent identically distributed with $B_i \in \{-1, 1\}$ and $\Pr\{B = 1\} = 1/2$, which are independent of the other random variables in the problem. These binary random variables represent the random action

selection of the random policy π_q. Then

$$\mathrm{E}_{\pi_{q_0},\ldots,\pi_{q_{j-1}},\pi_j,\ldots,\pi_T}\left\{\sum_{t=0}^{T} r(\mathbf{O}_t,\mathbf{A}_t,O_{t+1}\right\} =$$
$$\mathrm{E}\left\{\sum_{\mathbf{a}_T\in\{-1,1\}^{T+1}}\prod_{t=0}^{j-1} I\left(B_t=a_t\right)\prod_{t=j}^{T} I\left(\pi_t(\mathbf{O}_t^{\mathbf{a}_{t-1}},\mathbf{a}_{t-1})=a_t\right)\sum_{t=0}^{T} r(\mathbf{O}_t^{\mathbf{a}_{t-1}},\mathbf{a}_t,O_{t+1}^{\mathbf{a}_t})\right\}.$$

4.4 Problem Formulation

Consider a class of deterministic policies Π, i.e., each element of Π is a sequence of $T+1$ deterministic decision rules. It is possible to estimate the average value function $V(\pi)$ (4.2.6) of any policy in the class from the set of trajectory trees by simply averaging the sum of rewards on each tree along the path that agrees with the policy [62]. Denote by $\widehat{V}^i(\pi)$ the observed value on the i'th tree along the path that corresponds to the policy π. Then, the average value function of the policy π is estimated by its empirical average value function

$$\widehat{V}_n(\pi) = n^{-1}\sum_{i=1}^{n}\widehat{V}^i(\pi) =$$
$$\mathrm{E}_n\left\{\sum_{\mathbf{a}_T\in\{-1,1\}^{T+1}}\prod_{t=0}^{T} I\left(\pi_t(\mathbf{O}_t^{\mathbf{a}_{t-1}},\mathbf{a}^{t-1})=a_t\right)\sum_{t=0}^{T} r(\mathbf{O}_t^{\mathbf{a}_{t-1}},\mathbf{a}_t,O_{t+1}^{\mathbf{a}_t})\right\}. \quad (4.4.9)$$

In [62], the authors show that for policy classes with finite VC-dimension [5] (to be discussed below), with high probability over the data set, $\widehat{V}_n(\pi)$ converges uniformly over Π to $V(\pi)$ (4.2.6) with rates that depend on the VC-dimension of Π. This result motivates the use of policies π with high $\widehat{V}_n(\pi)$, since with high probability these policies have high values of $V(\pi)$. However, maximizing $\widehat{V}_n(\pi)$ over an infinite class of policies is computationally prohibited [6], [62] [12].

In this paper, we consider policy classes of the form $\Pi = \{(\pi_0,\pi_1,\ldots,\pi_T) : \pi_o \in$

$\Pi_0, \pi_1 \in \Pi_1, \ldots, \pi_T \in \Pi_T\}$, where for each t, Π_t is a class of single-stage decision rules, and tackle the problem of estimating the optimal policy based on a set of trajectory trees.

4.5 Binary Single-Stage Reinforcement Learning Problem

We begin with the single stage RL problem first, since it enables us to isolate the effect of finite data (Theorem 2) and surrogate optimization (Theorem 3) from the ones of distribution mismatch and error propagation that arise in multi-stage problems. The results in this section are then used to establish our two main results in the next section. Consider the following binary single-stage reinforcement learning problem. An agent observes $O_0 \in \mathcal{O}_0$ and can choose between one of two actions, $\mathcal{A} = \{-1, 1\}$. Upon choosing action $A_0 \in \mathcal{A}$ the agent observes O_1, which is also the final observation, and a reward $r(O_0, A_0, O_1)$ is received. Under the generative model assumption, it is possible to generate O_1 given any value of (O_0, A_0). Denote by (O_0, O_1^{-1}, O_1^1) the random variables generated by the generative model. That is, O_0 is generated according to the distribution of the initial observation, and given the value of O_0, O_1 is generated independently for the two possible actions, denoted by O_1^{-1} and O_1^1. The likelihood of a realization (o_0, o_1^{-1}, o_1^1) is given by

$$f_0(o_0) f_1(o_1^{-1}|o_0, -1) f_1(o_1^1|o_0, 1).$$

Denote an expectation with respect to this density by E, and given n realizations of (O_0, O_1^{-1}, O_1^1) denote by E_n the corresponding empirical expectation.

Let Π be a class of policies $\pi : \mathcal{O}_0 \to \mathcal{A}$. The average value function of a policy

π is given by (4.2.6)

$$V(\pi) = \mathrm{E}_\pi \left\{ r(O_0, A_0, O_1) \right\}.$$

As a special case of (4.4.9), it is possible to write this expectation with respect to E:

$$\begin{aligned}
&\mathrm{E}\left\{ r(O_0, -1, O_1^{-1}) I(\pi(O_0) = -1) + r(O_0, 1, O_1^1) I(\pi(O_0) = 1) \right\} \\
&= \int_{\mathcal{O}_0} \int_{\mathcal{O}_1} \int_{\mathcal{O}_1} \left[r(o_0, -1, o_1^{-1}) I(\pi(o_0) = -1) + r(o_0, 1, o_1^1) I(\pi(o_0) = 1) \right] \times \\
&f_0(o_0) f_1(o_1^{-1} | o_0, -1) f_1(o_1^1 | o_0, 1) do_0 do_1^{-1} do_1^1 \\
&= \int_{\mathcal{O}_0} \int_{\mathcal{O}_1} r(o_0, -1, o_1^{-1}) I(\pi(o_0) = -1) f_0(o_0) f_1(o_1^{-1} | -1) do_0 do_1^{-1} \\
&+ \int_{\mathcal{O}_0} \int_{\mathcal{O}_1} r(1, o_1^1) I(\pi(o_0) = 1) f_0(o_0) f_1(o_1^1 | o_0, 1) do_0 do_1^1 \\
&= \int_{\mathcal{O}_0} \left[\int_{\mathcal{O}_1} r(o_0, -1, o_1^{o_0, -1}) f_1(o_1^{-1} | o_0, -1) do_1^{-1} I(\pi(o_0) = -1) \right. \\
&\left. + \int_{\mathcal{O}_1} r(o_0, 1, o_1^1) f_1(o_1^1 | o_0, 1) do_1^1 I(\pi(o_0) = 1) \right] f_0(o_0) do_0 \\
&= \int_{\mathcal{O}_0} \left[\sum_{a_0 \in \mathcal{A}} \int_{\mathcal{O}_1} r(o_0, a_0, o_1^{a_0}) f_1(o_1^{a_0} | o_0, a_0) do_1^{a_0} I(\pi(o_0) = a_0) \right] f_0(o_0) do_0 \\
&= \int_{\mathcal{O}_0} \sum_{a_0 \in \mathcal{A}} \int_{\mathcal{O}_1} r(o_0, a_0, o_1) f_1(o_1 | o_0, a_0) I(\pi(o_0) = a_0) f_0(o_0) do_1 do_0 \\
&= \mathrm{E}_\pi \left\{ r(O_0, A_0, O_1) \right\}.
\end{aligned}$$

Therefore, it is possible to consistently estimate $V(\pi) = \mathrm{E}_\pi \left\{ r(O_0, A_0, O_1) \right\}$ by

$$\widehat{V}_n(\pi) = \mathrm{E}_n \left\{ r(O_0, -1, O_1^{-1}) I(\pi(O_0) = -1) + r(O_0, 1, O_1^1) I(\pi(O_0) = 1) \right\}.$$

Assuming that the sup is attainable, let $\widetilde{\pi}$ be any policy satisfying

$$V(\widetilde{\pi}) = \sup_{\pi \in \Pi} V(\pi).$$

As proposed in [62], let $\widehat{\pi}_n$ be any policy satisfying

$$\widehat{\pi}_n \in \arg\max_{\pi\in\Pi} \widehat{V}_n(\pi), \tag{4.5.10}$$

where the maximum exists since $\widehat{V}_n(\pi)$ can take a finite number of values.

To derive a finite sample upper bound on the performance of $\widehat{\pi}_n$ relative to those of $\widetilde{\pi}$, we need to restrict the size of the policy class Π. As originally applied to Markov decision processes in [62], we use the VC-dimension [5] to express this restriction. For a class Π of binary valued functions from space $\mathcal{X}$ to $\{-1, 1\}$ and a set $\{x^i\}_{i=1}^n \subset \mathcal{X}$ denote by $\Pi|_{\{x^i\}_{i=1}^n}$ the set $\{[\pi(x^1), \pi(x^2), \ldots, \pi(x^n)] | \pi \in \Pi\} \subset \{-1, 1\}^n$. We say that a set $\{x^i\}_{i=1}^n \subset \mathcal{X}$ is **shattered** by Π if the cardinality of $\Pi|_{\{x^i\}_{i=1}^n}$ is 2^n, where the cardinality of a set S, denoted by $|S|$, is the number of distinct elements.

Definition 1. *The* **VC-dimension** *of a class* Π *of binary valued functions from space* $\mathcal{X}$ *to* $\{-1, 1\}$ *is the largest* n*, for which there exists a set* $\{x^i\}_{i=1}^n \subset \mathcal{X}$ *of cardinality* n *that is shattered by* Π*. If no such number exists, we say that the VC-dimension of* Π *is infinity.*

Another interpretation is the following [5]. For any finite set $S \subset \mathcal{X}$, every function π in Π defines a dichotomy: $S_1 = \{x \in S : \pi(x) = 1\}$ and $S_{-1} = \{x \in S : \pi(x) = -1\}$, $S_{-1} \bigcup S_1 = S$, $S_{-1} \bigcap S_1 = \emptyset$. Then the VC-dimension of Π is the cardinality of the set with the largest number of elements for which members of Π can realize all possible dichotomies.

Given a collection of n realizations of the trajectory tree of the single-stage process, $\{o_0^i, o_1^{-1i}, o_1^{1i}\}_{i=1}^n \subset \mathcal{O}_0 \times \mathcal{O}_1 \times \mathcal{O}_1$, there are 2^n possible policy realizations, which correspond to all the combinations of taking action -1 or action 1 on each of the trees. When the policy class is a class of binary valued functions, that has a finite VC-dimension d, then, by Sauer's lemma [119], the number of possible policy

realizations grows with n in a polynomial rate, rather than as 2^n. This property is important for the uniform convergence results that we invoke below.

To account for the interaction between the binary functions that define the policy and the reward function we use the definition of the P-dimension [119], which generalizes the VC-dimension to real valued functions. Consider a class $\mathcal{F}$ of real valued functions from a space $\mathcal{X}$ to $[0, R]$. A set $S = \{x^1, x^2, \ldots, x^n\} \subset \mathcal{X}$ is said to be **P-shattered** by $\mathcal{F}$ if there exists a real vector $c \in [0, R]^n$, such that, for every binary vector $e \in \{-1, 1\}^n$, there exists a corresponding function $f_e \in \mathcal{F}$, such that $\text{sgn}\left(f_e(x^i) - c_i\right) = e_i$, $i = 1, 2, \ldots, n$, where $\text{sgn}(x)$ equals one if $x > 0$ and zero otherwise.

Definition 2. *The* **P-dimension** *of a class $\mathcal{F}$ of real valued functions from a space $\mathcal{X}$ to $[0, R]$ is the largest n for which there exists a set $\{x^i\}_{i=1}^n \subset \mathcal{X}$ of cardinality n that is P-shattered by $\mathcal{F}$. If no such n exists, we say the P-dimension of $\mathcal{F}$ is infinity.*

In the following theorem and in the other theorems in this chapter, the number of trajectory trees required to achieve a given performance guarantees is bounded by the VC-dimension or the P-dimension of the policy class. One approximation class for which bounds on the VC-dimension and the P-dimension exist is the class of Neural Networks. In [5] and [119] bounds on the VC-dimension and P-dimension a number of Neural Network architectures are given. The bounds depend on the number of layers, the size of the weights, and the form of the activation function.

Theorem 2. *Assume that* Π *has a finite VC-dimension d. Then with probability greater than $1 - \delta$ over the set of trajectory trees,*

$$V(\widetilde{\pi}) - V(\widehat{\pi}_n) \leq 2\epsilon$$

for n satisfying

$$4\left(\frac{8en}{\epsilon d}\right)^{d}\exp\left(\frac{-n\epsilon^{2}}{32}\right)\leq\delta.$$

Proof. For every $\pi\in\Pi$, define the reward $f:\mathcal{O}_0\times\mathcal{O}_1\times\mathcal{O}_1\rightarrow[0,1]$ by

$$f_{\pi}(o_0,o_1^{-1},o_1^{1})=r(o_0,-1,o_1^{-1})I(\pi(o_0)=-1)+r(o_0,1,o_1^{1})I(\pi(o_0)=1),$$

and let $\mathcal{F}=\{f_{\pi}:\pi\in\Pi\}$. We first show that the P-dimension of $\mathcal{F}$ is less than or equal to d and then invoke theorem 7.2 of [119]. Consider a set $S=\{o_0^{i},o_1^{-1i},o_1^{1i}\}_{i=1}^{m}\subset\mathcal{O}_0\times\mathcal{O}_1\times\mathcal{O}_1$ of m realizations of the trajectory tree of the underlying single stage decision process. For this set to be P-shattered by $\mathcal{F}$, we must have that the cardinality of the set

$$\mathcal{F}|_S=\{[f(o_0^{1},o_1^{-11},o_1^{11}),f(o_0^{1},o_1^{-12},o_1^{12}),\ldots,f(o_0^{m},o_1^{-1m},o_1^{1m})]:f\in\mathcal{F}\}\subset\mathbb{R}^{m}$$

is at least 2^m. Otherwise, comparing this set of vectors to a threshold vector $c\in\mathbb{R}^m$ cannot lead to the 2^m distinct binary vectors required for P-shattering. However, the cardinality of $\mathcal{F}|_S$ is bounded above by the cardinality of the set $\Pi|_S=\{[\pi(x^1),\pi(x^2),\ldots,\pi(x^m)]:\pi\in\Pi\}\subset\{-1,1\}^m$ since any distinct vector $[\pi(x^1),\pi(x^2),\ldots,\pi(x^m)]$ contributes at most one element to $\mathcal{F}|_S$ corresponding to the map defined by f_π. Note that we write 'at most' one element, since the cardinality of $\mathcal{F}|_S$ can actually be smaller than that of $\Pi|_S$ due to cases in which the reward is the same for both actions. For the cardinality of Π_S to be 2^m, m must be smaller than or equal to the VC-dimension of Π. Therefore the P-dimension of $\mathcal{F}$ is less than or equal to d.

By [119, Theorem 7.2] we have that with probability greater than $1-\delta$ over the

set of trajectory trees,

$$\sup_{f\in\mathcal{F}} |\mathrm{E}\left\{f(o_0, o_1^{-1}, o_1^1\right\} - \mathrm{E}_n\left\{f(o_0, o_1^{-1}, o_1^1\right\}| \leq \epsilon,$$

which is equivalent to

$$\sup_{\pi\in\Pi} |V(\pi) - \widehat{V}_n(\pi)| \leq \epsilon, \tag{4.5.11}$$

for n satisfying

$$4\left(\frac{8en}{\epsilon d}\right)^d \exp\left(\frac{-n\epsilon^2}{32}\right) \leq \delta. \tag{4.5.12}$$

This result (4.5.11) and (4.5.12) is a special case of [62, Theorem 3.2], in which only convergence rates are provided.

Result (4.5.11) and (4.5.12) imply that for such an n, with probability greater than $1-\delta$,

$$\begin{aligned} V(\widehat{\pi}_n) &> \widehat{V}_n(\widehat{\pi}_n) - \epsilon \\ &\geq \widehat{V}_n(\widetilde{\pi}) - \epsilon \\ &> V(\widetilde{\pi}) - 2\epsilon, \end{aligned}$$

where the first and third inequalities follow from (4.5.11), and the second inequality holds since $\widehat{\pi}_n$ maximizes $\widehat{V}_n$. The statement of the theorem follows. □

The theorem asserts that by minimizing the empirical value $\widehat{V}_n(\pi)$ one is guaranteed to find a policy whose performance are close to the best possible within the restricted class. However, the empirical risk minimization (4.5.10) is computationally demanding (see e.g. [6] and [12]) and can only be solved for small n and simple policy classes. As an alternative, we propose to reduce the reinforcement learning problem to a weighted classification problem and replace the problematic $0-1$ loss

with a convex smooth surrogate function.

Consider a class $\mathcal{F}$ of real valued functions $f : \mathcal{O}_0 \to [-1, 1]$. Each $f \in \mathcal{F}$ induces a policy $\pi_f(o_0) = \operatorname{sgn}[f(o_0)]$, $o_0 \in \mathcal{O}_0$. To formulate (4.5.10) as a weighted classification problem, note that

$$\begin{aligned} V(\pi_f) =& \mathrm{E}\left\{r(O_0, -1, O_1^{-1})I(\operatorname{sgn}[f(O_0)] = -1) + r(O_0, 1, O_1^1)I(\operatorname{sgn}(f(O_0)) = 1)\right\} \\ =& \mathrm{E}\left\{\max\{r(O_0, -1, O_1^{-1}), r(O_0, 1, O_1^1)\}\right\} - \\ & \mathrm{E}\left\{|r(O_0, -1, O_1^{-1}) - r(O_0, 1, O_1^1)| I\left(\operatorname{sgn}[f(O_0)] \neq \arg\max_{a \in \mathcal{A}} r(O_0, a, O_1^a)\right)\right\}. \end{aligned}$$

Therefore, (4.5.10) is equivalent to

$$\begin{aligned} \widehat{f}_n \in & \arg\min_{f \in \mathcal{F}} \mathrm{E}_n \left\{ |r(O_0, -1, O_1^{-1}) - r(O_0, 1, O_1^1)| \times \right. \\ & \left. I\left(\operatorname{sgn}[f(O_0)] \neq \arg\max_{a \in \mathcal{A}} r(O_0, a, O_1^a)\right) \right\} \\ = & \arg\min_{f \in \mathcal{F}} \mathrm{E}_n \left\{ |r(O_0, -1, O_1^{-1}) - r(O_0, 1, O_1^1)| I\left(\operatorname{sgn}[f(O_0)] \neq Y\right) \right\} \end{aligned} \tag{4.5.13}$$

where $Y = \arg\max_{a \in \mathcal{A}} r(O_0, a, O_1^a)$, which is a weighted classification problem with examples o_0^i, targets y^i, and weights $|r(o_0^i, -1, o_1^{-1i}) - r(o_0^i, 1, o_1^{1i})|$. Solving (4.5.13) is just as difficult as solving (4.5.10). For many function classes $\mathcal{F}$, however, it is much easier to solve

$$\widehat{f}_\phi^n \in \arg\min_{f \in \mathcal{F}} \mathrm{E}_n \left\{ |r(O_0, -1, O_1^{-1}) - r(O_0, 1, O_1^1)| \phi\left(f(O_0) Y\right) \right\}, \tag{4.5.14}$$

where ϕ is a convex surrogate for the $0 - 1$ loss. For example, one can minimize a truncated squared error loss by using neural networks [23], an exponential loss by using Boosting [45], the scaled deviance using logistic regression [46], and the hinge loss by using support vector machines [103]

Below we show that, for restricted classes $\mathcal{F}$, a uniform convergence result can guarantee that

$$\begin{aligned}&\left|\mathrm{E}\left\{|r(O_0,-1,O_1^{-1})-r(O_0,1,O_1^1)|\phi\left(f(O_0)Y\right)\right\}-\right.\\&\left.\mathrm{E}_n\left\{|r(O_0,-1,O_1^{-1})-r(O_0,1,O_1^1)|\phi\left(f(O_0)Y\right)\right\}\right|\to 0\end{aligned}\tag{4.5.15}$$

almost surely, as $n\to\infty$, uniformly over $\mathcal{F}$. This implies that with high probability, for sufficiently large n, $\widehat{f}_\phi^n$ given by (4.5.14) is close to

$$\widetilde{f}_\phi=\arg\min_{f\in\mathcal{F}}\mathrm{E}\left\{|r(O_0,-1,O_1^{-1})-r(O_0,1,O_1^1)|\phi\left(f(O_0)Y\right)\right\},$$

where we assume that the minimum exists. Alternatively, this implies that

$$\mathrm{E}\left\{|r(O_0,-1,O_1^{-1})-r(O_0,1,O_1^1)|\phi\left(\widehat{f}_\phi^n(O_0)Y\right)\right\}$$

is close to

$$\mathrm{E}\left\{|r(O_0,-1,O_1^{-1})-r(O_0,1,O_1^1)|\phi\left(\widetilde{f}_\phi(O_0)Y\right)\right\},$$

with high probability over the data set. First, we apply the result in [8] to show that this also implies that $V(\pi_{\widehat{f}_\phi^n})$ is close to $V(\pi^*)$ with high probability. Then we prove uniform convergence of the type (4.5.15).

Note that minimizing

$$\mathrm{E}\left\{|r(O_0,-1,O_1^{-1})-r(O_0,1,O_1^1)|\phi\left(f(O_0)Y\right)\right\}$$

is equivalent to minimizing

$$\mathrm{E}\left\{\frac{|r(O_0,-1,O_1^{-1})-r(O_0,1,O_1^1)|}{\mathrm{E}\left\{|r(O_0,-1,O_1^{-1})-r(O_0,1,O_1^1)|\right\}}\phi\left(f(O_0)Y\right)\right\},$$

since the constant

$$c = \mathrm{E}\left\{|r(O_0, -1, O_1^{-1}) - r(O_0, 1, O_1^1)|\right\}$$

does not depend on f, and we assume that $c \neq 0$. This is equivalent to minimizing

$$\widetilde{\mathrm{E}}\left\{\phi\left(f(O_0)Y\right)\right\},$$

where $\widetilde{\mathrm{E}}$ is the expectation with respect to the distribution induced by the change of measure associated with the multiplication by $\frac{|r(O_0,-1,O_1^{-1})-r(O_0,1,O_1^1)|}{\mathrm{E}\left\{|r(O_0,-1,O_1^{-1})-r(O_0,1,O_1^1)|\right\}}$. Note that $c \leq 1$ and that

$$\min_f \widetilde{\mathrm{E}}\left\{I(\mathrm{sgn}[f(O_0)] \neq Y\right\} = \widetilde{\mathrm{E}}\left\{I\left(\pi^*(O_0) \neq Y\right)\right\},$$

since the optimal policy π^* is invariant to change of measure.

Applying the result in [8] we obtain that if the surrogate loss ϕ is convex, differentiable at 0, and $\phi'(0) < 0$, conditions that hold for all the algorithms mentioned above, then, for any function f,

$$\begin{aligned}
&\psi\left(\widetilde{\mathrm{E}}\left\{I(\mathrm{sgn}[f(O_0)] \neq Y\right\} - \widetilde{\mathrm{E}}\left\{I(\pi^*(O_0) \neq Y\right\}\right) \\
&\leq \widetilde{\mathrm{E}}\left\{\phi\left(f(O_0)Y\right)\right\} - \widetilde{\mathrm{E}}\left\{\phi\left(f_\phi^*(O_0)Y\right)\right\},
\end{aligned}$$

where ψ is a convex function that can be derived from ϕ, invertible on $[0, 1]$, $\psi(0) = 0$, and, assuming the minimum exists, we let

$$f_\phi^* = \arg\min_f \mathrm{E}\left\{|r(O_0, -1, O_1^{-1}) - r(O_0, 1, O_1^1)|\phi\left(f(O_0)Y\right)\right\}.$$

Furthermore, it is shown that this bound is the tightest possible without placing

further restriction on the underlying distribution. For a convex function $g : \mathbb{R} \to \mathbb{R}$ with $g(0) = 0$, it holds that for any $\lambda \in [0, 1]$ and $x \in \mathbb{R}$, $g(\lambda x) \leq \lambda g(x)$ (see [101] or [8]). Therefore,

$$
\begin{aligned}
&\psi\left(\mathrm{E}\left\{|r(O_0, -1, O_1^{-1}) - r(O_0, 1, O_1^1)| I\left(\mathrm{sgn}[f(O_0)] \neq Y\right)\right\} - \right. \\
&\left. \mathrm{E}\left\{|r(O_0, -1, O_1^{-1}) - r(O_0, 1, O_1^1)| I\left(\pi^*(O_0) \neq Y\right)\right\}\right) = \\
&\psi\left(c\left[\widetilde{\mathrm{E}}\left\{I\left(\mathrm{sgn}[f(O_0)] \neq Y\right)\right\} - \widetilde{\mathrm{E}}\left\{I\left(\pi^*(O_0) \neq Y\right)\right\}\right]\right) \leq \\
&c\psi\left(\widetilde{\mathrm{E}}\left\{I\left(\mathrm{sgn}[f(O_0)] \neq Y\right)\right\} - \widetilde{\mathrm{E}}\left\{I\left(\pi^*(O_0) \neq Y\right)\right\}\right) \leq \\
&c\left[\widetilde{\mathrm{E}}\left\{\phi\left(f(O_0)Y\right)\right\} - \widetilde{\mathrm{E}}\left\{\phi\left(f_\phi^*(O_0)Y\right)\right\}\right] = \\
&\mathrm{E}\left\{|r(O_0, -1, O_1^{-1}) - r(O_0, 1, O_1^1)| \phi\left(f(O_0)Y\right)\right\} - \\
&\mathrm{E}\left\{|r(O_0, -1, O_1^{-1}) - r(O_0, 1, O_1^1)| \phi\left(f_\phi^*(O_0)Y\right)\right\}
\end{aligned}
$$

The invertibility of ψ on $[0, 1]$ implies that

$$
\begin{aligned}
&\mathrm{E}\left\{|r(O_0, -1, O_1^{-1}) - r(O_0, 1, O_1^1)| I(\mathrm{sgn}[f(O_0)] \neq Y\right\} - \\
&\mathrm{E}\left\{|r(O_0, -1, O_1^{-1}) - r(O_0, 1, O_1^1)| I(\pi^*(O_0) \neq Y\right\} \\
&\leq \psi^{-1}\left(\mathrm{E}\left\{|r(O_0, -1, O_1^{-1}) - r(O_0, 1, O_1^1)| \phi\left(f(O_0)Y\right)\right\} - \right. \\
&\left. \mathrm{E}\left\{|r(O_0, -1, O_1^{-1}) - r(O_0, 1, O_1^1)| \phi\left(f_\phi^*(O_0)Y\right)\right\}\right).
\end{aligned}
$$

This result implies that

$$
\begin{aligned}
&V(\pi^*) - V(\pi_{\widehat{f}^n_\phi}) = \\
&\mathrm{E}\left\{\max\{r(O_0,-1,O_1^{-1}), r(O_0,1,O_1^1)\}\right\} - \\
&\mathrm{E}\left\{|r(O_0,-1,O_1^{-1}) - r(O_0,1,O_1^1)| I\left(\pi^*(O_0) \neq Y\right)\right\} - \\
&\mathrm{E}\left\{\max\{r(O_0,-1,O_1^{-1}), r(O_0,1,O_1^1)\}\right\} + \\
&\mathrm{E}\left\{|r(O_0,-1,O_1^{-1}) - r(O_0,1,O_1^1)| I\left(\mathrm{sgn}[\widehat{f}^n_\phi(O_0)] \neq Y\right)\right\} \\
&\leq \psi^{-1}\left(\mathrm{E}\left\{|r(O_0,-1,O_1^{-1}) - r(O_0,1,O_1^1)| \phi\left(\widehat{f}^n_\phi(O_0) Y\right)\right\} - \right. \\
&\left.\mathrm{E}\left\{|r(O_0,-1,O_1^{-1}) - r(O_0,1,O_1^1)| \phi\left(f^*_\phi(O_0) Y\right)\right\}\right). \qquad (4.5.16)
\end{aligned}
$$

Next rewrite the argument of ψ^{-1} in the upper bound as

$$
\begin{aligned}
&\mathrm{E}\left\{|r(O_0,-1,O_1^{-1}) - r(O_0,1,O_1^1)| \phi\left(\widehat{f}^n_\phi(O_0) Y\right)\right\} - \\
&\mathrm{E}\left\{|r(O_0,-1,O_1^{-1}) - r(O_0,1,O_1^1)| \phi\left(f^*_\phi(O_0) Y\right)\right\} = \\
&\mathrm{E}\left\{|r(O_0,-1,O_1^{-1}) - r(O_0,1,O_1^1)| \phi\left(\widehat{f}^n_\phi(O_0) Y\right)\right\} - \\
&\inf_{f\in\mathcal{F}} \mathrm{E}\left\{|r(O_0,-1,O_1^{-1}) - r(O_0,1,O_1^1)| \phi\left(f(O_0) Y\right)\right\} + \\
&\inf_{f\in\mathcal{F}} \mathrm{E}\left\{|r(O_0,-1,O_1^{-1}) - r(O_0,1,O_1^1)| \phi\left(f(O_0) Y\right)\right\} - \\
&\mathrm{E}\left\{|r(O_0,-1,O_1^{-1}) - r(O_0,1,O_1^1)| \phi\left(f^*_\phi(O_0) Y\right)\right\} \qquad (4.5.17)
\end{aligned}
$$

The first term is called the estimation error and the second term is called the approximation error, which we denote by $\gamma(\mathcal{F})$. Assuming the inf is attainable, let

$$
\begin{aligned}
&\mathrm{E}\left\{|r(O_0,-1,O_1^{-1}) - r(O_0,1,O_1^1)| \phi\left(\widetilde{f}_\phi(O_0) Y\right)\right\} = \\
&\inf_{f\in\mathcal{F}} \mathrm{E}\left\{|r(O_0,-1,O_1^{-1}) - r(O_0,1,O_1^1)| \phi\left(f(O_0) Y\right)\right\}
\end{aligned}
$$

for some $\widetilde{f}_\phi \in \mathcal{F}$.

Before stating the following theorem, which provides a finite sample upper bound on the estimation error, we need to define the external covering number of a set. Define the average 1-norm on $\mathbb{R}^n$ by $||v||_{a1} = \frac{1}{n}\sum_{i=1}^{n}|v_i|$. A set $v^1, v^2, \ldots, v^k$ is an external ϵ-cover with respect to the average 1-norm of a set $S \subset \mathbb{R}^n$ if for any $u \in S$ there exists v^j such that $||u - v^j||_{a1} < \epsilon$. It is called an external ϵ-cover since the vectors $v^1, v^2, \ldots, v^k$ need not be in S. The cardinality of the set with the smallest number of elements that is an external ϵ-cover of $\mathcal{F}|_{\{x^i\}_{i=1}^n}$ with respect to the average 1-norm is called the **external ϵ-covering number** of $\mathcal{F}|_{\{x^i\}_{i=1}^n}$ with respect to the average 1-norm and denoted by $L(\epsilon, \mathcal{F}|_{\{x^i\}_{i=1}^n}, ||\cdot||_{a1})$.

Theorem 3. *Suppose $\mathcal{F}$ has finite P-dimension d, and that ϕ satisfies a Lipschitz condition*

$$|\phi(x_1) - \phi(x_2)| \leq \mu|x_1 - x_2|, \quad x_1, x_2 \in [-1, 1] \tag{4.5.18}$$

for some μ. Then with probability greater than $1 - \delta$ over the set of trajectory trees,

$$V(\pi^*) - V(\pi_{\widehat{f}_\phi^n}) \leq \psi^{-1}\left(2\epsilon + \gamma(\mathcal{F})\right)$$

for n satisfying

$$8\left(\frac{16e\mu}{\epsilon}\ln\frac{16e\mu}{\epsilon}\right)^d \exp(-m\epsilon^2/32) < \delta.$$

Proof. The proof is based on a uniform convergence result which is very similar to Theorem 7.5 of [119], where the only difference is in the form of the loss function. Let $\mathcal{L}_\mathcal{F}$ be the class of functions mapping $\mathcal{O}_0 \times \mathcal{O}_1 \times \mathcal{O}_1 \times \{-1, 1\}$ to $[0, 1]$ defined by

$$\mathcal{L}_\mathcal{F} = \{l_f(o_0, o_1^{-1}, o_1^1, y) = |r(o_0, -1, o_1^{-1}) - r(o_0, 1, o_1^1)|\phi(f(o_0)y) : f \in \mathcal{F}\}.$$

To apply the proof of Theorem 7.5 in [119] we need to bound the external covering number of $\mathcal{L}_{\mathcal{F}}$ in terms of the covering number of $\mathcal{F}$.

Given the set $S = \{o_0^i, o_1^{-1i}, o_1^{1i}, y^i\}_{i=1}^{2n} \subset \mathcal{O}_0 \times \mathcal{O}_1 \times \mathcal{O}_1 \times \{-1, 1\}$, we show that

$$L(\epsilon, \mathcal{L}_{\mathcal{F}}|_{\{o_0^i, o_1^{-1i}, o_1^{1i}, y^i\}_{i=1}^{2n}}, ||\cdot||_{a1}) \leq L(\epsilon/\mu, \mathcal{F}|_{\{o_0^i\}_{i=1}^{2n}}, ||\cdot||_{a1}). \tag{4.5.19}$$

Suppose $(v^1, v^2, \ldots, v^k)$, $v^j \in \mathbb{R}^{2n}$ is an external ϵ/μ cover for $\mathcal{F}|_{\{o_0^i\}_{i=1}^{2n}}$. We show that the set of k vectors $(w^1, w^2, \ldots, w^k)$, $w^j \in \mathbb{R}^{2n}$ defined by

$$w_i^j = |r(o_0^i, -1, o_1^{-1i}) - r(o_0, 1, o_1^{1i})|\phi(v_i^j y^i), \quad 1 \leq i \leq 2n, \quad 1 \leq j \leq k,$$

is an external ϵ cover for $\mathcal{L}_{\mathcal{F}}|_{\{o_0^i, o_1^{-1i}, o_1^{1i}, y^i\}_{i=1}^{2n}}$, which implies (4.5.19). To see this consider an arbitrary $l_f \in \mathcal{L}_{\mathcal{F}}$. Since $(v^1, v^2, \ldots, v^k)$, $v^j \in \mathbb{R}^{2n}$ is an external ϵ/μ cover for $\mathcal{F}|_{\{o_0^i\}_{i=1}^{2n}}$, there exists an index j for which

$$||f|_{\{o_0^i\}_{i=1}^{2n}} - v^j||_{a1} = \frac{1}{2n} \sum_{i=1}^{2n} |f(o_0^i) - v_i^j| \leq \epsilon/\mu,$$

where $f|_{\{o_0^i\}_{i=1}^{2n}} = [f(o_0^1), f(o_0^2), \ldots, f(o_0^{2n})]$. The Lipschitz condition implies that

$$
\begin{aligned}
&||l_f|_{\{o_0^i, o_1^{-1i}, o_1^{1i}, y^i\}_{i=1}^{2n}} - w^j||_{a1} = \\
&\frac{1}{2n}\sum_{i=1}^{2n} \Big| |r(o_0^i, -1, o_1^{-1i}) - r(o_0^i, 1, o_1^{1i})| \times \\
&\qquad \phi(f(o_0^i)y^i) - |r(o_0^i, -1, o_1^{-1i}) - r(o_0^i, 1, o_1^{1i})|\phi(v_i^j y^i) \Big| = \\
&\frac{1}{2n}\sum_{i=1}^{2n} |r(o_0^i, -1, o_1^{-1i}) - r(o_0^i, 1, o_1^{1i})| \times |\phi(f(o_0^i)y^i) - \phi(v_i^j y^i)| \leq \\
&\frac{1}{2n}\sum_{i=1}^{2n} |\phi(f(o_0^i)y^i) - \phi(v_i^j y^i)| \leq \\
&\mu\frac{1}{2n}\sum_{i=1}^{2n} |f(o_0^i)y^i - v_i^j y^j| = \\
&\mu\frac{1}{2n}\sum_{i=1}^{2n} |f(o_0^i) - v_i^j| \leq \epsilon,
\end{aligned}
$$

where the first inequality holds since $r \in [0, 1]$, which is what we need to show. Hence for any $2n$-set we have

$$
L(\epsilon, \mathcal{L}_{\mathcal{F}}|_{\{o_0^i, o_1^{-1i}, o_1^{1i}, y^i\}_{i=1}^{2n}}, ||\cdot||_{a1}) \leq L(\epsilon/\mu, \mathcal{F}|_{\{o_0^i\}_{i=1}^{2n}}, ||\cdot||_{a1}) \leq 2\left(\frac{2\mathrm{e}\mu}{\epsilon}\ln\frac{2\mathrm{e}\mu}{\epsilon}\right)^d,
$$

where the second inequality follows from [119, Corollary 4.2] since $\mathcal{F}$ has P-dimension d. Given this bound we follow the steps in [119] to obtain that with probability greater than $1 - \delta$ over the set of trajectory trees,

$$
\sup_{f\in\mathcal{F}} |\mathrm{E}\left\{|r(S_1^{-1}) - r(S_1^1)|\phi(f(S_0)Y)\right\} - \mathrm{E}_n\left\{|r(S_1^{-1}) - r(S_1^1)|\phi(f(S_0)Y)\right\}| < \epsilon
$$

for n satisfying

$$
8\left(\frac{16\mathrm{e}\mu}{\epsilon}\ln\frac{16\mathrm{e}\mu}{\epsilon}\right)^d \exp(-m\epsilon^2/32) < \delta.
$$

Next note that since $\widehat{f}_\phi^n$ is the minimizer of the empirical expectation,

$$\begin{aligned}
&\mathrm{E}_n\left\{|r(O_0,-1,O_1^{-1})-r(O_0,1,O_1^1)|\phi\left(\widetilde{f}_\phi(O_0)Y\right)\right\}-\\
&\mathrm{E}_n\left\{|r(O_0,-1,O_1^{-1})-r(O_0,1,O_1^1)|\phi\left(\widehat{f}_\phi^n(O_0)Y\right)\right\}
\end{aligned}$$

is greater than or equal to zero. Therefore,

$$\begin{aligned}
&\mathrm{E}\left\{|r(O_0,-1,O_1^{-1})-r(O_0,1,O_1^1)|\phi\left(\widehat{f}_\phi^n(O_0)Y\right)\right\}-\\
&\mathrm{E}\left\{|r(O_0,-1,O_1^{-1})-r(O_0,1,O_1^1)|\phi\left(\widetilde{f}_\phi(O_0)Y\right)\right\}\leq\\
&\left|\mathrm{E}\left\{|r(O_0,-1,O_1^{-1})-r(O_0,1,O_1^1)|\phi\left(\widehat{f}_\phi^n(O_0)Y\right)\right\}-\right.\\
&\left.\mathrm{E}_n\left\{|r(O_0,-1,O_1^{-1})-r(O_0,1,O_1^1)|\phi\left(\widehat{f}_\phi^n(O_0)Y\right)\right\}\right|+\\
&\left|\mathrm{E}\left\{|r(O_0,-1,O_1^{-1})-r(O_0,1,O_1^1)|\phi\left(\widetilde{f}_\phi(O_0)Y\right)\right\}-\right.\\
&\left.\mathrm{E}_n\left\{|r(O_0,-1,O_1^{-1})-r(O_0,1,O_1^1)|\phi\left(\widetilde{f}_\phi(O_0)Y\right)\right\}\right|\leq 2\epsilon
\end{aligned}$$

with probability greater than $1-\delta$, and the statement of the theorem follows from (4.5.16) and (4.5.17). □

When comparing Theorem 3 with Theorem 2 it is seen that the computational advantage of the surrogate optimization has a cost: we can no longer guarantee that as the number of samples increase the estimated policy approaches the best within the approximation class. Only when $\gamma(\mathcal{F})=0$, this type of consistency holds.

4.6 An Approximate Dynamic Programming Approach

This section generalize the results of the previous section to the multi-stage case. We describe an approximate dynamic programming algorithm for approximating $\widehat{\pi}$ (4.2.7), (4.2.8) from a set of n trajectory trees and derive finite sample upper bounds on its generalization error. The algorithm estimates $\widehat{\pi}_T, \widehat{\pi}_{T-1}, \ldots, \widehat{\pi}_0$ sequentially, starting from

$$\widehat{\pi}_T^n \in \tag{4.6.20}$$
$$\arg\max_{\pi_T \in \Pi_T} \mathrm{E}_n \left\{ \sum_{\mathbf{a}_T \in \{-1,1\}^{T+1}} \prod_{t=0}^{T-1} I\left(B_t = a_t\right) \times \right.$$
$$\left. I\left(\pi_T(\mathbf{O}_T^{\mathbf{a}_{T-1}}, \mathbf{a}_{T-1}) = a_T\right) r(\mathbf{O}_T^{\mathbf{a}_{T-1}}, \mathbf{a}_T, O_{T+1}^{\mathbf{a}_T}) \right\},$$

which can be performed by randomly selecting a leaf at stage T from each trajectory tree and solving the single stage reinforcement learning algorithm from time T to $T+1$. Given $\widehat{\pi}_T^n, \widehat{\pi}_{T-1}^n, \ldots, \widehat{\pi}_{t+1}^n$, we can form the empirical counter part of (4.2.8), by choosing a random leaf at stage t from each tree and considering the immediate reward following each of the actions plus the reward accumulated by following decision rules $\widehat{\pi}_{t+1}^n, \widehat{\pi}_{t+2}^n, \ldots, \widehat{\pi}_T^n$ for stages $t+1$ and on. That is,

$$\widehat{\pi}_t^n \in \arg\max_{\pi_t \in \Pi_t} \tag{4.6.21}$$
$$\mathrm{E}_n \left\{ \sum_{\mathbf{a}_T \in \{-1,1\}^{T+1}} \prod_{t=0}^{t-1} I\left(B_t = a_t\right) I\left(\pi_t(\mathbf{O}_t^{\mathbf{a}_{t-1}}, \mathbf{a}_{t-1}) = a_t\right) \times \right.$$
$$\left. \prod_{\tau=t+1}^{T} I\left(\widehat{\pi}_\tau^n(\mathbf{O}_\tau^{\mathbf{a}_{\tau-1}}, \mathbf{a}_{\tau-1}) = a_\tau\right) \sum_{t=0}^{T} r(\mathbf{O}_\tau^{\mathbf{a}_{\tau-1}}, \mathbf{a}_\tau, O_{\tau+1}^{\mathbf{a}_\tau}) \right\}.$$

Next a finite sample bound on the generalization error of $\widehat{\pi}^n = (\widehat{\pi}_0^n, \widehat{\pi}_1^n, \ldots, \widehat{\pi}_T^n)$ is derived. The following lemmas will be useful in deriving the bound. For $i \leq j$, we use the shorthand notation $\pi_{q_{i,\ldots,j}}$ for $(\pi_{q_i}, \ldots, \pi_{q_j})$, $\pi_{i,\ldots,j}$ for $\pi_i, \ldots, \pi_j$, and similarly for π^* and $\widehat{\pi}$. The lemma are derived first in terms of distribution over random trajectories $\mathbf{O}_{T+1}, \mathbf{A}_T$ rather than in terms of random variables on the trajectory tree since the notation is more manageable this way. Then, the results are translated to expectations with respect to E, i.e., with respect to random variables on the trajectory tree.

The following lemmas are used to relate the results of the previous section for the single-stage problem to the multi-stage problem. The observation distribution in (4.2.7), and (4.2.8) is induced by a random policy rather than by the optimal policy. The following lemma relates the expectation of a positive function with respect to these to distributions.

Lemma 6. *Fix $0 \leq j \leq T$. For a random policy $\pi_p = \pi_{p_0}, \pi_{p_1}, \ldots, \pi_{p_T}$ with $p_t(a|\mathbf{o}_t, \mathbf{a}_{t-1}) \geq 1/\overline{L}$, for all $a \in \mathcal{A}$ and $0 \leq t \leq T$, a deterministic policy π, and a function $f(\mathbf{o}_{T+1}, \mathbf{a}_T) \geq 0$, we have*

$$\mathrm{E}_{\pi_{p_{0,\ldots,j}}, \pi_{j+1,\ldots,T}} \{f(\mathbf{O}_{T+1}, \mathbf{A}_T)\} \leq \overline{L} \mathrm{E}_{\pi_{p_{0,\ldots,j-1}}, \pi_{j,\ldots,T}} \{f(\mathbf{O}_{T+1}, \mathbf{A}_T)\}$$

Proof. The integration in $\mathrm{E}_{\pi_{p_{0,\ldots,j}}, \pi_{j+1,\ldots,T}} \{f(\mathbf{O}_{T+1}, \mathbf{A}_T)\}$ is with respect to the density

$$f_0(o_0) p_0(a_0|o_0) \prod_{t=1}^{j} f_t(o_t|\mathbf{o}_{t-1}, \mathbf{a}_{t-1}) p_t(a_t|\mathbf{o}_t, \mathbf{a}_{t-1}) \times$$
$$\prod_{t=j+1}^{T} f_t(o_t|\mathbf{o}_{t-1}, \mathbf{a}_{t-1}) I(a_t = \pi_t(\mathbf{o}_t, \mathbf{a}_{t-1})) f_T(o_{T+1}|\mathbf{o}_T, \mathbf{a}_T).$$

The integration in $\mathrm{E}_{\pi_{p_0,\ldots,j-1},\pi_{j,\ldots,T}}\{f(\mathbf{O}_{T+1},\mathbf{A}_T)\}$ is with respect to the density

$$f_0(o_0)p_0(a_0|o_0)\prod_{t=1}^{j-1}f_t(o_t|\mathbf{o}_{t-1},\mathbf{a}_{t-1})p_t(a_t|\mathbf{o}_t,\mathbf{a}_{t-1})\times$$
$$\prod_{t=j}^{T}f_t(o_t|\mathbf{o}_{t-1},\mathbf{a}_{t-1})I(a_t=\pi_t(\mathbf{o}_t,\mathbf{a}_{t-1}))f_T(o_{T+1}|\mathbf{o}_T,\mathbf{a}_T).$$

Hence

$$\mathrm{E}_{\pi_{p_0,\ldots,j},\pi_{j+1,\ldots,T}}\{f(\mathbf{O}_{T+1},\mathbf{A}_T)\}=$$
$$\mathrm{E}_{\pi_{p_0,\ldots,j-1},\pi_{j,\ldots,T}}\left\{\frac{I(a_j=\pi_j(\mathbf{o}_j,\mathbf{a}_{j-1}))}{p_j(a_j|\mathbf{o}_j,\mathbf{a}_{j-1})}f(\mathbf{O}_{T+1},\mathbf{A}_T)\right\}.$$

The result follows since $0\leq\frac{I(a_j=\pi_j(\mathbf{o}_j,\mathbf{a}_{j-1}))}{p_j(a_j|\mathbf{o}_j,\mathbf{a}_{j-1})}\leq\overline{L}$ and f is non-negative. □

The next lemma is helpful for analyzing the consequences of using $\widehat{\pi}^n_{t+1},\ldots,\widehat{\pi}^n_T$ rather than the optimal policy when estimating $\widehat{\pi}^n_t$.

Lemma 7. *For the random policy $\pi_q=(\pi_{q_0},\ldots,\pi_{q_T})$, the optimal policy π^*, and the policy $\widehat{\pi}$, we have for $j=0,\ldots,T-1$,*

$$\mathrm{E}_{\pi_{q_0,\ldots,j-1},\pi^*_{j,\ldots,T}}\left\{\sum_{t=j}^{T}r(\mathbf{O}_t,\mathbf{A}_t,O_{t+1})\right\}$$
$$\leq\mathrm{E}_{\pi_{q_0,\ldots,j-1},\pi^*_j,\widehat{\pi}_{j+1,\ldots,T}}\left\{\sum_{t=j}^{T}r(\mathbf{O}_j,\mathbf{A}_j,O_{j+1})\right\}$$
$$+2\mathrm{E}_{\pi_{q_0,\ldots,j},\pi^*_{j+1,\ldots,T}}\left\{\sum_{t=j+1}^{T}r(\mathbf{O}_t,\mathbf{A}_t,O_{t+1})\right\}$$
$$-2\mathrm{E}_{\pi_{q_0,\ldots,j},\widehat{\pi}_{j+1,\ldots,T}}\left\{\sum_{t=j+1}^{T}r(\mathbf{O}_t,\mathbf{A}_t,O_{t+1})\right\}$$

Proof. The proof follows from the previous lemma and the use of conditional expec-

tations.

$$
\begin{aligned}
&\mathrm{E}_{\pi_{q_{0,\ldots,j-1}},\pi^*_{j,\ldots,T}}\left\{\sum_{t=j}^{T} r(\mathbf{O}_t,\mathbf{A}_t,O_{t+1})\right\}\\
&=\mathrm{E}_{\pi_{q_{0,\ldots,j-1}},\pi^*_{j,\ldots,T}}\left\{r(\mathbf{O}_j,\mathbf{A}_j,O_{j+1})+\sum_{t=j+1}^{T} r(\mathbf{O}_t,\mathbf{A}_t,O_{t+1})\right\}\\
&=\mathrm{E}_{\pi_{q_{0,\ldots,j-1}},\pi^*_{j,\ldots,T}}\Bigg\{r(\mathbf{O}_j,\mathbf{A}_j,O_{j+1})+\\
&\qquad\mathrm{E}_{\pi_{q_{0,\ldots,j-1}},\pi^*_{j,\ldots,T}}\left\{\sum_{t=j+1}^{T} r(\mathbf{O}_t,\mathbf{A}_t,O_{t+1})\Big|\mathbf{O}_{j+1},\mathbf{A}_j\right\}\Bigg\}\\
&=\mathrm{E}_{\pi_{q_{0,\ldots,j-1}},\pi^*_{j,\ldots,T}}\left\{r(\mathbf{O}_j,\mathbf{A}_j,O_{j+1})+V_{(\cdot,\pi^*_{j+1,\ldots,T}),j+1}(\mathbf{O}_{j+1},\mathbf{A}_j)\right\}\\
&=\mathrm{E}_{\pi_{q_{0,\ldots,j-1}},\pi^*_{j,\ldots,T}}\left\{r(\mathbf{O}_j,\mathbf{A}_j,O_{j+1})+V_{(\cdot,\widehat{\pi}_{j+1,\ldots,T}),j+1}(\mathbf{O}_{j+1},\mathbf{A}_j)\right\}\\
&+\mathrm{E}_{\pi_{q_{0,\ldots,j-1}},\pi^*_{j,\ldots,T}}\left\{V_{(\cdot,\pi^*_{j+1,\ldots,T}),j+1}(\mathbf{O}_{j+1},\mathbf{A}_j)-V_{(\cdot,\widehat{\pi}_{j+1,\ldots,T}),j+1}(\mathbf{O}_{j+1},\mathbf{A}_j)\right\}
\end{aligned}
$$

Proceed with the first term:

$$
\begin{aligned}
&\mathrm{E}_{\pi_{q_{0,\ldots,j-1}},\pi^*_{j,\ldots,T}}\left\{r(\mathbf{O}_j,\mathbf{A}_j,O_{j+1})+V_{(\cdot,\widehat{\pi}_{j+1,\ldots,T}),j+1}(\mathbf{O}_{j+1},\mathbf{A}_j)\right\}\\
&=\mathrm{E}_{\pi_{q_{0,\ldots,j-1}},\pi^*_{j},\widehat{\pi}_{j+1,\ldots,T}}\left\{r(\mathbf{O}_j,\mathbf{A}_j,O_{j+1})+V_{(\cdot,\widehat{\pi}_{j+1,\ldots,T}),j+1}(\mathbf{O}_{j+1},\mathbf{A}_j)\right\}\\
&=\mathrm{E}_{\pi_{q_{0,\ldots,j-1}},\pi^*_{j},\widehat{\pi}_{j+1,\ldots,T}}\Bigg\{r(\mathbf{O}_j,\mathbf{A}_j,O_{j+1})+\\
&\qquad\mathrm{E}_{(\cdot,\widehat{\pi}_{j+1,\ldots,T})}\left\{\sum_{t=j+1}^{T} r(\mathbf{O}_t,\mathbf{A}_t,O_{t+1}|\mathbf{O}_{j+1},\mathbf{A}_j\right\}\Bigg\}\\
&=\mathrm{E}_{\pi_{q_{0,\ldots,j-1}},\pi^*_{j},\widehat{\pi}_{j+1,\ldots,T}}\Bigg\{r(\mathbf{O}_j,\mathbf{A}_j,O_{j+1})+\\
&\qquad\mathrm{E}_{\pi_{q_{0,\ldots,j-1}},\pi^*_{j},\widehat{\pi}_{j+1,\ldots,T}}\left\{\sum_{t=j+1}^{T} r(\mathbf{O}_t,\mathbf{A}_t,O_{t+1}|\mathbf{O}_{j+1},\mathbf{A}_j\right\}\Bigg\}\\
&=\mathrm{E}_{\pi_{q_{0,\ldots,j-1}},\pi^*_{j},\widehat{\pi}_{j+1,\ldots,T}}\left\{\sum_{t=j}^{T} r(\mathbf{O}_j,\mathbf{A}_j,O_{j+1})\right\},
\end{aligned}
$$

where the first inequality holds since the function

$$r(\mathbf{O}_j, \mathbf{A}_j, O_{j+1}) + V_{(\cdot,\widehat{\pi}_{j+1,\ldots,T}),j+1}(\mathbf{O}_{j+1}, \mathbf{A}_j)$$

does not depend on the actions and observations following O_{t+1} and so the decision rules for actions $A_{t+1}, \ldots, A_T$ can be specified arbitrarily. The second equality follows from the definition of $V_{(\cdot,\widehat{\pi}_{j+1,\ldots,T}),j+1}(\mathbf{O}_{j+1}, \mathbf{A}_j)$. The third inequality holds since, due to the conditioning on $\mathbf{O}_{j+1}$ and $\mathbf{A}_j$, it is possible to specify the decision rules for $A_0, \ldots, A_j$ arbitrarily. The fourth equality follows from the properties of the conditional expectation. As for the second term:

$$\begin{aligned}
&\mathrm{E}_{\pi_{q_{0,\ldots,j-1}},\pi^*_{j,\ldots,T}} \left\{ V_{(\cdot,\pi^*_{j+1,\ldots,T}),j+1}(\mathbf{O}_{j+1}, \mathbf{A}_j) - V_{(\cdot,\widehat{\pi}_{j+1,\ldots,T}),j+1}(\mathbf{O}_{j+1}, \mathbf{A}_j) \right\} \\
&\leq 2\mathrm{E}_{\pi_{q_{0,\ldots,j}},\pi^*_{j+1,\ldots,T}} \left\{ V_{(\cdot,\pi^*_{j+1,\ldots,T}),j+1}(\mathbf{O}_{j+1}, \mathbf{A}_j) - V_{(\cdot,\widehat{\pi}_{j+1,\ldots,T}),j+1}(\mathbf{O}_{j+1}, \mathbf{A}_j) \right\}
\end{aligned}$$

by Lemma 6 with $\overline{L} = 2$, since

$$V_{(\cdot,\pi^*_{j+1,\ldots,T}),j+1}(\mathbf{O}_{j+1}, \mathbf{A}_j) - V_{(\cdot,\widehat{\pi}_{j+1,\ldots,T}),j+1}(\mathbf{O}_{j+1}, \mathbf{A}_j)$$

is non-negative by the definition of π^*. Next,

$$\begin{aligned}
&\mathrm{E}_{\pi_{q_{0,\ldots,j}},\pi^*_{j+1,\ldots,T}} \left\{ V_{(\cdot,\pi^*_{j+1,\ldots,T}),j+1}(\mathbf{O}_{j+1}, \mathbf{A}_j) \right\} \\
&= \mathrm{E}_{\pi_{q_{0,\ldots,j}},\pi^*_{j+1,\ldots,T}} \left\{ \mathrm{E}_{\pi_{q_{0,\ldots,j}},\pi^*_{j+1,\ldots,T}} \left\{ \sum_{t=j+1}^{T} r(\mathbf{O}_t, \mathbf{A}_t, O_{t+1}) | \mathbf{O}_{j+1}, \mathbf{A}_j \right\} \right\} \\
&= \mathrm{E}_{\pi_{q_{0,\ldots,j}},\pi^*_{j+1,\ldots,T}} \left\{ \sum_{t=j+1}^{T} r(\mathbf{O}_t, \mathbf{A}_t, O_{t+1}) \right\}
\end{aligned}$$

and

$$
\begin{aligned}
&\mathrm{E}_{\pi_{q_{0,\ldots,j}},\pi^*_{j+1,\ldots,T}}\left\{V_{(\cdot,\widehat{\pi}_{j+1,\ldots,T}),j+1}(\mathbf{O}_{j+1},\mathbf{A}_j)\right\}\\
&=\mathrm{E}_{\pi_{q_{0,\ldots,j}},\widehat{\pi}_{j+1,\ldots,T}}\left\{V_{(\cdot,\widehat{\pi}_{j+1,\ldots,T}),j+1}(\mathbf{O}_{j+1},\mathbf{A}_j)\right\}\\
&=\mathrm{E}_{\pi_{q_{0,\ldots,j}},\widehat{\pi}_{j+1,\ldots,T}}\left\{\mathrm{E}_{\pi_{q_{0,\ldots,j}},\widehat{\pi}_{j+1,\ldots,T}}\left\{\sum_{t=j+1}^{T} r(\mathbf{O}_t,\mathbf{A}_t,O_{t+1})|\mathbf{O}_{j+1},\mathbf{A}_j\right\}\right\}\\
&=\mathrm{E}_{\pi_{q_{0,\ldots,j}},\widehat{\pi}_{j+1,\ldots,T}}\left\{\sum_{t=j+1}^{T} r(\mathbf{O}_t,\mathbf{A}_t,O_{t+1})\right\}.
\end{aligned}
$$

Combining these together, we obtain the result. □

In the following lemma the difference between the value of a given policy and the value of the optimal policy is expresses in terms that are minimized by the approximate dynamic programming algorithm (4.6.20), (4.6.21).

Lemma 8. *For the policies π_q, π^*, and $\widehat{\pi}$, we have*

$$
\begin{aligned}
&V(\pi^*)-V(\widehat{\pi})\leq\\
&\sum_{\tau=0}^{T} 2^{\tau}\left[\mathrm{E}_{\pi_{q_{0,\ldots,\tau-1}},\pi^*_{\tau},\widehat{\pi}_{\tau+1,\ldots,T}}\left\{\sum_{t=\tau}^{T} r(\mathbf{O}_t,\mathbf{A}_t,O_{t+1})\right\}-\right.\\
&\left.\mathrm{E}_{\pi_{q_{0,\ldots,\tau-1}},\widehat{\pi}_{\tau,\ldots,T}}\left\{\sum_{t=\tau}^{T} r(\mathbf{O}_t,\mathbf{A}_t,O_{t+1})\right\}\right].
\end{aligned}
$$

Proof. By definition

$$
V(\pi^*)-V(\widehat{\pi})=\mathrm{E}_{\pi^*}\left[\sum_{t=0}^{T} r(\mathbf{O}_t,\mathbf{A}_t,O_{t+1})\right]-\mathrm{E}_{\widehat{\pi}}\left[\sum_{t=0}^{T} r(\mathbf{O}_t,\mathbf{A}_t,O_{t+1})\right].
$$

Applying Lemma 7 on the first term once we get

$$
\begin{aligned}
V(\pi^*) - V(\widehat{\pi}) &= \mathrm{E}_{\pi^*}\left[\sum_{t=0}^{T} r(\mathbf{O}_t, \mathbf{A}_t, O_{t+1})\right] - \mathrm{E}_{\widehat{\pi}}\left[\sum_{t=0}^{T} r(\mathbf{O}_t, \mathbf{A}_t, O_{t+1})\right] \\
&\leq \mathrm{E}_{\pi_0^*, \widehat{\pi}_{1\ldots,T}}\left[\sum_{t=0}^{T} r(\mathbf{O}_t, \mathbf{A}_t, O_{t+1})\right] - \mathrm{E}_{\widehat{\pi}}\left[\sum_{t=0}^{T} r(\mathbf{O}_t, \mathbf{A}_t, O_{t+1})\right] \\
&+ 2\mathrm{E}_{\pi_{p_0}, \pi^*_{1,\ldots,T}}\left[\sum_{t=1}^{T} r(\mathbf{O}_t, \mathbf{A}_t, O_{t+1})\right] - 2\mathrm{E}_{\pi_{p_0}, \widehat{\pi}_{1,\ldots,T}}\left[\sum_{t=1}^{T} r(\mathbf{O}_t, \mathbf{A}_t, O_{t+1})\right].
\end{aligned}
$$

Applying it again on the third term we get

$$
\begin{aligned}
&\leq \mathrm{E}_{\pi_0^*, \widehat{\pi}_{1\ldots,T}}\left[\sum_{t=0}^{T} r(\mathbf{O}_t, \mathbf{A}_t, O_{t+1})\right] - \mathrm{E}_{\widehat{\pi}}\left[\sum_{t=0}^{T} r(\mathbf{O}_t, \mathbf{A}_t, O_{t+1})\right] \\
&+ 2\mathrm{E}_{\pi_{p_0}, \pi^*_1, \widehat{\pi}_{2,\ldots,T}}\left[\sum_{t=1}^{T} r(\mathbf{O}_t, \mathbf{A}_t, O_{t+1})\right] - 2\mathrm{E}_{\pi_{p_0}, \widehat{\pi}_{1,\ldots,T}}\left[\sum_{t=1}^{T} r(\mathbf{O}_t, \mathbf{A}_t, O_{t+1})\right] \\
&+ 2^2\mathrm{E}_{\pi_{p_{0,1}}, \pi^*_{2,\ldots,T}}\left[\sum_{t=2}^{T} r(\mathbf{O}_t, \mathbf{A}_t, O_{t+1})\right] - 2^2\mathrm{E}_{\pi_{p_{0,1}}, \widehat{\pi}_{2,\ldots,T}}\left[\sum_{t=2}^{T} r(\mathbf{O}_t, \mathbf{A}_t, O_{t+1})\right].
\end{aligned}
$$

Applying it additional $T-2$ times we get

$$
\begin{aligned}
&V(\pi^*) - V(\widehat{\pi}) \\
&\leq \mathrm{E}_{\pi_0^*, \widehat{\pi}_{1\ldots,T}} \left[\sum_{t=0}^{T} r(\mathbf{O}_t, \mathbf{A}_t, O_{t+1}) \right] - \mathrm{E}_{\widehat{\pi}} \left[\sum_{t=0}^{T} r(\mathbf{O}_t, \mathbf{A}_t, O_{t+1}) \right] \\
&+2\mathrm{E}_{\pi_{q_0}, \pi_1^*, \widehat{\pi}_{2,\ldots,T}} \left[\sum_{t=1}^{T} r(\mathbf{O}_t, \mathbf{A}_t, O_{t+1}) \right] - 2\mathrm{E}_{\pi_{q_0}, \widehat{\pi}_{1,\ldots,T}} \left[\sum_{t=1}^{T} r(\mathbf{O}_t, \mathbf{A}_t, O_{t+1}) \right] \\
&+2^2\mathrm{E}_{\pi_{q_{0,1}}, \pi_2^*, \widehat{\pi}_{3,\ldots,T}} \left[\sum_{t=2}^{T} r(\mathbf{O}_t, \mathbf{A}_t, O_{t+1}) \right] - 2^2\mathrm{E}_{\pi_{q_{0,1}}, \widehat{\pi}_{2,\ldots,T}} \left[\sum_{t=2}^{T} r(\mathbf{O}_t, \mathbf{A}_t, O_{t+1}) \right] \\
&\vdots \\
&+2^{T-1}\mathrm{E}_{\pi_{q_{0,\ldots,T-2}}, \pi_{T-1}^*, \widehat{\pi}_T} \left[\sum_{t=T-1}^{T} r(\mathbf{O}_t, \mathbf{A}_t, O_{t+1}) \right] \\
&-2^{T-1}\mathrm{E}_{\pi_{q_{0,\ldots,T-2}}, \widehat{\pi}_{T-1,T}} \left[\sum_{t=T-1}^{T} r(\mathbf{O}_t, \mathbf{A}_t, O_{t+1}) \right] \\
&+2^T\mathrm{E}_{\pi_{q_{0,\ldots,T-1}}, \pi_T^*} \left[r(\mathbf{O}_T, \mathbf{A}_T, O_{t+1}) \right] - 2^T\mathrm{E}_{\pi_{q_{0,\ldots,T-1}}, \widehat{\pi}_T} \left[r(\mathbf{O}_T, \mathbf{A}_T, O_{T+1}) \right],
\end{aligned}
$$

which is the result. □

Next, we tie the expectations to the empirical expectations used to estimate the policy $\widehat{\pi}^n$. First recall that for any j

$$
\begin{aligned}
&\mathrm{E}_{\pi_{q_0}, \ldots, \pi_{q_{j-1}}, \pi_j, \ldots, \pi_T} \left\{ \sum_{t=0}^{T} r(\mathbf{O}_t, \mathbf{A}_t, O_{t+1} \right\} = \\
&\mathrm{E} \left\{ \sum_{\mathbf{a}_T \in \{-1,1\}^{T+1}} \prod_{t=0}^{j-1} I\left(B_t = a_t\right) \prod_{t=j}^{T} I\left(\pi_t(\mathbf{O}_t^{\mathbf{a}_{t-1}}, \mathbf{a}^{t-1}) = a_t\right) \sum_{t=j}^{T} r(\mathbf{O}_t^{\mathbf{a}_{t-1}}, \mathbf{a}^t, O_{t+1}^{\mathbf{a}_t}) \right\}
\end{aligned}
$$

which can be simplified to

$$\mathrm{E}\left\{\sum_{[a_j,\ldots,a_T]\in\{-1,1\}^{T+1-j}}\prod_{t=j}^{T} I\left(\pi_t(\mathbf{O}_t^{[\mathbf{B}_{j-1},\mathbf{a}_{j,t-1}]},[\mathbf{B}_{j-1},\mathbf{a}_{j,t-1}])=a_t\right)\times \sum_{t=j}^{T} r(\mathbf{O}_t^{[\mathbf{B}_{j-1},\mathbf{a}_{j,t-1}]},[\mathbf{B}_{j-1},\mathbf{a}_{j,t}],O_{t+1}^{[\mathbf{B}_{j-1},\mathbf{a}_{j,t}]})\right\},$$

where $\mathbf{a}_{j,t} = (a_j, a_{j+1}, \ldots, a_t)$. Therefore, the previous lemma can be expressed in terms of expectations with respect to the elements on the trajectory tree. Through an abuse of notation we use the shorthand notation

$$\gamma_{\pi_{q_{0,j-1}},\pi_{j,\ldots,T}}(\mathbf{O}_{T+1},\mathbf{B}_T)$$

for the argument in the last expectation.

Lemma 9. *For the policies π_q, π^*, and π, we have*

$$V(\pi^*) - V(\widehat{\pi}) \leq \sum_{\tau=0}^{T} 2^\tau \left[\mathrm{E}\left\{\gamma_{\pi_{q_{0,\tau-1}},\pi^*_\tau,\widehat{\pi}_{\tau+1,\ldots,T}}(\mathbf{O}_{T+1},\mathbf{B}_T)\right\} - \mathrm{E}\left\{\gamma_{\pi_{q_{0,\tau-1}},\widehat{\pi}_{\tau,\ldots,T}}(\mathbf{O}_{T+1},\mathbf{B}_T)\right\}\right]$$

Now we relate the difference between the average value functions to the distance between ensemble expectations and empirical expectations, using the trick in [84, page 1087].

Lemma 10. *For $\tau = 0, \ldots, T$,*

$$\begin{aligned}
&\mathrm{E}\left\{\gamma_{\pi_{q_{0,\tau-1}},\pi^*_\tau,\widehat{\pi}_{\tau+1,\ldots,T}}(\mathbf{O}_{T+1},\mathbf{B}_T)\right\} - \mathrm{E}\left\{\gamma_{\pi_{q_{0,\tau-1}},\widehat{\pi}_{\tau,\ldots,T}}(\mathbf{O}_{T+1},\mathbf{B}_T)\right\} \\
&\leq \left|\mathrm{E}\left\{\gamma_{\pi_{q_{0,\tau-1}},\pi^*_\tau,\widehat{\pi}_{\tau+1,\ldots,T}}(\mathbf{O}_{T+1},\mathbf{B}_T)\right\} - \mathrm{E}_n\left\{\gamma_{\pi_{q_{0,\tau-1}},\pi^*_\tau,\widehat{\pi}_{\tau+1,\ldots,T}}(\mathbf{O}_{T+1},\mathbf{B}_T)\right\}\right| \\
&+ \left|\mathrm{E}\left\{\gamma_{\pi_{q_{0,\tau-1}},\widehat{\pi}_{\tau,\ldots,T}}(\mathbf{O}_{T+1},\mathbf{B}_T)\right\} - \mathrm{E}_n\left\{\gamma_{\pi_{q_{0,\tau-1}},\widehat{\pi}_{\tau,\ldots,T}}(\mathbf{O}_{T+1},\mathbf{B}_T)\right\}\right| \\
&+ \left(\mathrm{E}_n\left\{\gamma_{\pi_{q_{0,\tau-1}},\pi^*_\tau,\widehat{\pi}_{\tau+1,\ldots,T}}(\mathbf{O}_{T+1},\mathbf{B}_T)\right\} - \mathrm{E}_n\left\{\gamma_{\pi_{q_{0,\tau-1}},\widehat{\pi}_{\tau,\ldots,T}}(\mathbf{O}_{T+1},\mathbf{B}_T)\right\}\right)^+,
\end{aligned}$$

where $(x)^+$ is the positive part of x.

Note that if $\pi^*_\tau \in \Pi_\tau$ and we replace $\widehat{\pi}$ with $\widehat{\pi}^n$ then the last term is zero since $\widehat{\pi}^n_\tau$ is the maximizer of

$$\mathrm{E}_n\left\{\gamma_{\pi_{q_{0,\tau-1}},\pi_\tau,\widehat{\pi}^n_{\tau+1,\ldots,T}}(\mathbf{O}_{T+1},\mathbf{B}_T)\right\}$$

over Π_τ.

The following lemma is a uniform convergence result that we then use to bound the terms in Lemma 9.

Lemma 11. *If the VC-dimension of the classes $\Pi_\tau, \Pi_{\tau+1}, \ldots, \Pi_T$ are $d_\tau, d_{\tau+1}, \ldots, d_T$, respectively, then, the probability over the set of trajectory trees that*

$$\sup_{\pi_\tau\in\Pi_\tau,\ldots,\pi_T\in\Pi_T}\left|\mathrm{E}\left\{\gamma_{\pi_{q_{0,\tau-1}},\pi_\tau,\ldots,T}(\mathbf{O}_{T+1},\mathbf{B}_T)\right\} - \mathrm{E}_n\left\{\gamma_{\pi_{q_{0,\tau-1}},\pi_\tau,\ldots,T}(\mathbf{O}_{T+1},\mathbf{B}_T)\right\}\right| > \epsilon$$

is less then or equal to

$$4\left(\frac{8en(T-\tau+1)}{\epsilon d_{sum,\tau}}\right)^{d_{sum,\tau}}\exp\left(-\frac{n\epsilon^2}{32}\right),$$

where $d_{sum,\tau} = \sum_{t=\tau}^T d_t$.

Remark 5. *If $\pi^* \notin \Pi$ but one wishes to include π^* in the above* sup*, then the bound holds if we replace $d_{sum,\tau}$ with $\overline{d}_{sum,\tau} = \sum_{t=\tau}^{T}(d_t + 1)$.*

Proof. The result follows from Theorem 7.2 of [119]. To apply the theorem we need to compute a bound on the P-dimension of the associated function class. For any $\pi_\tau \in \Pi_\tau, \ldots, \pi_T \in \Pi_T$, let $h_{\pi_\tau,\ldots,\pi_T}$ be

$$h_{\pi_1,\ldots,\pi_T}(G, B_0, \ldots, B_{\tau-1}) = \sum_{\mathbf{a}_T \in \{-1,1\}^{T+1}} \prod_{t=0}^{\tau-1} I\left(B_t = a_t\right) \prod_{t=\tau}^{T} I\left(\pi_t(\mathbf{O}_t^{\mathbf{a}_{t-1}}, \mathbf{a}^{t-1}) = a_t\right) \sum_{t=\tau}^{T} r(\mathbf{O}_t^{\mathbf{a}_{t-1}}, \mathbf{a}^t, O_{t+1}^{\mathbf{a}_t})$$

and let

$$\mathcal{H} = \left\{h_{\pi_1,\ldots,\pi_T} : \pi_1 \in \Pi_1, \ldots, \pi_T \in \Pi_T\right\}.$$

In order for $\mathcal{H}$ to P-shatter a set of size m, functions from $\mathcal{H}$ must realize at least 2^m values when realized on this set. However, the number of possible realizations is bounded by the product of the numbers of realizations of each of the indicator functions on the same set. If $m \geq \max\{d_1, \ldots, d_T\}$, then by Sauer's lemma [119] this product is bounded by $\prod_{t=1}^{T}(em/d_t)^{d_t} \leq (em/d_{min})^{d_{sum}}$, where $d_{sum} = \sum_{t=1}^{T} d_t$ and $d_{min} = \min\{d_1, \ldots, d_T\}$. Hence,

$$m < d_{sum} \log_2(em/d_{min})$$

Clearly, any $m > d_{sum}$ does not satisfy this inequality. Therefore, $m < d_{sum}$, implying that the P-dimension of $\mathcal{H}$ is less the or equal to d_{sum}. Finally, note that functions in $\mathcal{H}$ take values in $[0, T - \tau + 1]$. The lemma now follows from [119, Theorem 7.2]. □

Using the union bound we can bound the probability of a large difference between

ensemble and empirical means for some $0 \leq \tau \leq T$:

$$\begin{aligned}
&\Pr\Bigg\{ \bigcup_{\tau=0}^{T} \sup_{\pi_\tau \in \Pi_\tau, \ldots, \pi_T \in \Pi_T} \Big| \mathrm{E}\left\{ \gamma_{\pi_{q_{0,\tau-1}}, \pi_\tau, \ldots, T}(\mathbf{O}_{T+1}, \mathbf{B}_T) \right\} \\
&\qquad\qquad - \mathrm{E}\left\{ \gamma_{\pi_{q_{0,\tau-1}}, \pi_\tau, \ldots, T}(\mathbf{O}_{T+1}, \mathbf{B}_T) \right\} \Big| > \epsilon \Bigg\} \\
&\leq \sum_{\tau=0}^{T} \Pr\Bigg\{ \sup_{\pi_\tau \in \Pi_\tau, \ldots, \pi_T \in \Pi_T} \Big| \mathrm{E}\left\{ \gamma_{\pi_{q_{0,\tau-1}}, \pi_\tau, \ldots, T}(\mathbf{O}_{T+1}, \mathbf{B}_T) \right\} \\
&\qquad\qquad - \mathrm{E}\left\{ \gamma_{\pi_{q_{0,\tau-1}}, \pi_\tau, \ldots, T}(\mathbf{O}_{T+1}, \mathbf{B}_T) \right\} \Big| > \epsilon \Bigg\} \\
&\leq \sum_{\tau=0}^{T} 4 \left(\frac{8en(T-\tau+1)}{\epsilon d_{sum,\tau}} \right)^{d_{sum,\tau}} \exp\left(-\frac{n\epsilon^2}{32} \right).
\end{aligned}$$

This is equivalent to the statement: with probability greater than $1-\delta$ over the set of trajectory trees,

$$\sup_{\pi_\tau \in \Pi_\tau, \ldots, \pi_T \in \Pi_T} \Big| \mathrm{E}\left\{ \gamma_{\pi_{q_{0,\tau-1}}, \pi_\tau, \ldots, T}(\mathbf{O}_{T+1}, \mathbf{B}_T) \right\} - \mathrm{E}\left\{ \gamma_{\pi_{q_{0,\tau-1}}, \pi_\tau, \ldots, T}(\mathbf{O}_{T+1}, \mathbf{B}_T) \right\} \Big| < \epsilon$$

simultaneously for all τ, for n satisfying

$$\sum_{\tau=0}^{T} 4 \left(\frac{8en(T-\tau+1)}{\epsilon d_{sum,\tau}} \right)^{d_{sum,\tau}} \exp\left(-\frac{n\epsilon^2}{32} \right) < \delta. \tag{4.6.22}$$

This directly leads to the following theorem.

Theorem 4. *Let* $\Pi = \{(\pi_0, \pi_1, \ldots, \pi_T) : \pi_0 \in \Pi_0, \ldots, \pi_T \in \Pi_T\}$ *be a class of deterministic policies with* $VC - dim(\Pi_t) = d_t$, $t = 0, 1 \ldots, T$. *If* $\pi^* \in \Pi$ *then with probability greater than* $1-\delta$ *over a set of random trajectory trees,*

$$V(\pi^*) - V(\widehat{\pi}^n) \leq 2\epsilon \sum_{\tau=0}^{T} 2^\tau$$

for n satisfying (4.6.22).

If $\pi^* \notin \Pi$ then with probability greater than $1-\delta$ over a set of random trajectory trees,

$$V(\pi^*) - V(\widehat{\pi}^n) \leq 2\epsilon \sum_{\tau=0}^{T} 2^\tau$$
$$+ \sum_{\tau=0}^{T} 2^\tau \left(\mathrm{E}_n \left\{ \gamma_{\pi_{q_{0,\tau-1}}, \pi^*_\tau, \widehat{\pi}^n_{\tau+1,\ldots,T}}(\mathbf{O}_{T+1}, \mathbf{B}_T) \right\} - \mathrm{E}_n \left\{ \gamma_{\pi_{q_{0,\tau-1}}, \widehat{\pi}^n_{\tau,\ldots,T}}(\mathbf{O}_{T+1}, \mathbf{B}_T) \right\} \right)^+$$

for n satisfying (4.6.22) with $d_{sum,\tau}$ replaced with $\overline{d}_{sum,\tau}$.

Remark 6. *One can upper bound the term*

$$\mathrm{E}_n \left\{ \gamma_{\pi_{q_{0,\tau-1}}, \pi^*_\tau, \widehat{\pi}^n_{\tau+1,\ldots,T}}(\mathbf{O}_{T+1}, \mathbf{B}_T) \right\},$$

which cannot be computed, by $\max_{\pi_\tau} \mathrm{E}_n \left\{ \gamma_{\pi_{q_{0,\tau-1}}, \pi_\tau, \widehat{\pi}^n_{\tau+1,\ldots,T}}(\mathbf{O}_{T+1}, \mathbf{B}_T) \right\}$. This term can be easily computed by averaging the cumulative reward resulting from taking the maximizing action at stage τ on the randomly chosen leaf on every tree, given that the actions at the following stages are taken according to $\widehat{\pi}^n_{\tau+1}, \widehat{\pi}^n_{\tau+2}, \ldots, \widehat{\pi}^n_T$.

As mentioned earlier, when the policy class is large, it is difficult to solve

$$\widehat{\pi}^n_\tau \in \arg\max_{\pi_\tau \in \Pi_\tau} \mathrm{E}_n \left\{ \gamma_{\pi_{q_{0,\tau-1}}, \pi_\tau, \widehat{\pi}^n_{\tau+1,\ldots,T}}(\mathbf{O}_{T+1}, \mathbf{B}_T) \right\}. \tag{4.6.23}$$

Next, it is shown that (4.6.23) can be solved via the reduction to classification approach presented in Sec. 4.5, i.e., by introducing a class of real valued functions and solving a sequence of single stage reinforcement learning problems through a weighted classification reduction and using a surrogate ϕ for the $0-1$ loss. Then, a finite sample upper bound for the performance of a policy estimated in this manner is given.

Consider the classes $\mathcal{F}_t$, $t = 0, 1, \ldots, T$ of functions mapping possible histories $(\mathbf{o}_t, \mathbf{a}_{t-1})$ to $[-1, 1]$, $t = 0, 1, \ldots, T$, respectively. Assume each $\mathcal{F}_t$ has a finite P-dimension and denote these by $d_0, d_1, \ldots, d_T$. Each of the function classes induces a policy class; for $f_t \in \mathcal{F}_t$, $\pi_t(f_t)(\mathbf{o}_t, \mathbf{a}_{t-1}) = \operatorname{sgn}(f(\mathbf{o}_t, \mathbf{a}_{t-1}))$, $t = 0, \ldots, T$. Let $\Pi_t(\mathcal{F}_t) = \{\pi_t(f_t) : f_t \in \mathcal{F}_t\}$, $t = 0, 1, \ldots, T$. Note that by [119, Lemma 10.1], $\mathrm{VC} - \dim(\Pi_t(\mathcal{F}_t)) \leq d_t$, $t = 0, 1, \ldots, T$.

We start with the last stage. First we write the optimization problem

$$\widehat{f}_T^n \in \arg\max_{f_T \in \mathcal{F}_T} \mathrm{E}_n \left\{ \gamma_{\pi_{q_{0,T-1}}, \pi_T(f_T)}(\mathbf{O}_{T+1}, \mathbf{B}_T) \right\}. \tag{4.6.24}$$

as a weighted classification problem.

$$\begin{aligned}
&\mathrm{E}\left\{ \gamma_{\pi_{q_{0,T-1}}, \pi_T(f_T)}(\mathbf{O}_{T+1}, \mathbf{B}_T) \right\} = \\
&\mathrm{E}\left\{ \sum_{a_T \in \{-1,1\}} I\left(\pi_T(f_T)(\mathbf{O}_T^{\mathbf{B}_{T-1}}, \mathbf{B}_{T-1}) = a_T \right) r\left(\mathbf{O}_T^{\mathbf{B}_{T-1}}, [\mathbf{B}_{T-1}, a_T], O_{t+1}^{[\mathbf{B}_{T-1}, a_T]} \right) \right\} = \\
&\mathrm{E}\left\{ I\left(\pi_T(f_T)(\mathbf{O}_T^{\mathbf{B}_{T-1}}, \mathbf{B}_{T-1}) = -1 \right) r\left(\mathbf{O}_T^{\mathbf{B}_{T-1}}, [\mathbf{B}_{T-1}, -1], O_{t+1}^{[\mathbf{B}_{T-1}, -1]} \right) + \right. \\
&\left. I\left(\pi_T(f_T)(\mathbf{O}_T^{\mathbf{B}_{T-1}}, \mathbf{B}_{T-1}) = 1 \right) r\left(\mathbf{O}_T^{\mathbf{B}_{T-1}}, [\mathbf{B}_{T-1}, 1], O_{t+1}^{[\mathbf{B}_{T-1}, 1]} \right) \right\} = \\
&\mathrm{E}\left\{ \max_{a_T \in \{-1,1\}} \left\{ r\left(\mathbf{O}_T^{\mathbf{B}_{T-1}}, [\mathbf{B}_{T-1}, a_T], O_{t+1}^{[\mathbf{B}_{T-1}, a_T]} \right) \right\} \right\} - \\
&\mathrm{E}\left\{ \left| r\left(\mathbf{O}_T^{\mathbf{B}_{T-1}}, [\mathbf{B}_{T-1}, -1], O_{t+1}^{[\mathbf{B}_{T-1}, -1]} \right) - r\left(\mathbf{O}_T^{\mathbf{B}_{T-1}}, [\mathbf{B}_{T-1}, 1], O_{t+1}^{[\mathbf{B}_{T-1}, 1]} \right) \right| \times \right. \\
&\left. I\left(\pi_T(f_T)(\mathbf{O}_T^{\mathbf{B}_{T-1}}, \mathbf{B}_{T-1}) \neq \arg\max_{a_T \in \{-1,1\}} \left\{ r\left(\mathbf{O}_T^{\mathbf{B}_{T-1}}, [\mathbf{B}_{T-1}, a_T], O_{t+1}^{[\mathbf{B}_{T-1}, a_T]} \right) \right\} \right) \right\}
\end{aligned}$$

Therefore, (4.6.24) is equivalent to

$$\widehat{f}_T^n \in \arg\min_{f_T \in \mathcal{F}_T} \mathrm{E}_n \Bigg\{ \Big| r\left(\mathbf{O}_T^{\mathbf{B}_{T-1}}, [\mathbf{B}_{T-1}, -1], O_{t+1}^{[\mathbf{B}_{T-1}, -1]}\right) - r\left(\mathbf{O}_T^{\mathbf{B}_{T-1}}, [\mathbf{B}_{T-1}, 1], O_{t+1}^{[\mathbf{B}_{T-1}, 1]}\right) \Big| \times I\left(\pi_T(f_T)(\mathbf{O}_T^{\mathbf{B}_{T-1}}, \mathbf{B}_{T-1}) \neq \arg\max_{a_T \in \{-1,1\}} \left\{ r\left(\mathbf{O}_T^{\mathbf{B}_{T-1}}, [\mathbf{B}_{T-1}, a_T], O_{t+1}^{[\mathbf{B}_{T-1}, a_T]}\right)\right\}\right) \Bigg\}.$$

By defining the random variable

$$Y = \arg\max_{a_T \in \{-1,1\}} \left\{ r\left(\mathbf{O}_T^{\mathbf{B}_{T-1}}, [\mathbf{B}_{T-1}, a_T], O_{t+1}^{[\mathbf{B}_{T-1}, a_T]}\right)\right\},$$

we can write the optimization as a weighted classification problem

$$A\widehat{f}_T^n \in \arg\min_{f_T \in \mathcal{F}_T} \mathrm{E}_n \Bigg\{ \Big| r\left(\mathbf{O}_T^{\mathbf{B}_{T-1}}, [\mathbf{B}_{T-1}, -1], O_{t+1}^{[\mathbf{B}_{T-1}, -1]}\right) - r\left(\mathbf{O}_T^{\mathbf{B}_{T-1}}, [\mathbf{B}_{T-1}, 1], O_{t+1}^{[\mathbf{B}_{T-1}, 1]}\right) \Big| \times I\left(\pi_T(f_T)(\mathbf{O}_T^{\mathbf{B}_{T-1}}, \mathbf{B}_{T-1}) \neq Y\right) \Bigg\}.$$

Finally, introducing the surrogate ϕ for the $0-1$ loss we obtain

$$\widehat{f}_{\phi T}^n \in \arg\min_{f_T \in \mathcal{F}_T} \mathrm{E}_n \Bigg\{ \Big| r\left(\mathbf{O}_T^{\mathbf{B}_{T-1}}, [\mathbf{B}_{T-1}, -1], O_{t+1}^{[\mathbf{B}_{T-1}, -1]}\right) - r\left(\mathbf{O}_T^{\mathbf{B}_{T-1}}, [\mathbf{B}_{T-1}, 1], O_{t+1}^{[\mathbf{B}_{T-1}, 1]}\right) \Big| \times \phi\left(f_T(\mathbf{O}_T^{\mathbf{B}_{T-1}}, \mathbf{B}_{T-1}) Y\right) \Bigg\},$$

which is often a more feasible optimization problem.

For $\tau = T-1, \ldots, 0$, given $\widehat{f}^n_{\phi T}, \widehat{f}^n_{\phi T-1}, \ldots, \widehat{f}^n_{\phi \tau+1}$, we write

$$\mathrm{E}\left\{\gamma_{\pi_{q_{0,\ldots,\tau-1}},\pi_\tau(f_\tau),\pi_{t+1}(\widehat{f}^n_{\phi\tau+1}),\ldots,\pi_T(\widehat{f}^n_{\phi T})}(\mathbf{O}_{T+1},\mathbf{B}_T)\right\}$$

explicitly as

$$\begin{aligned}
&\mathrm{E}\left\{\sum_{[a_\tau,\ldots,a_T]\in\{-1,1\}^{T+1-\tau}} I\left(\pi_\tau(f_\tau)(\mathbf{O}_\tau^{\mathbf{B}_{\tau-1}},\mathbf{B}_{\tau-1})=a_\tau\right)\right.\\
&\prod_{t=\tau+1}^{T} I\left(\pi_t(\widehat{f}^n_{\phi t})\left(\mathbf{O}_t^{[\mathbf{B}_{\tau-1},\mathbf{a}_{\tau,t-1}]},[\mathbf{B}_{\tau-1},\mathbf{a}_{\tau,t-1}]\right)=a_t\right)\times\\
&\left.\sum_{t=\tau}^{T} r\left(\mathbf{O}_t^{[\mathbf{B}_{\tau-1},\mathbf{a}_{\tau,t-1}]},[\mathbf{B}_{\tau-1},\mathbf{a}_{\tau,t}],O_{t+1}^{[\mathbf{B}_{\tau-1},\mathbf{a}_{\tau,t}]}\right)\right\}=\\
&\mathrm{E}\left\{\, I\left(\pi_\tau(f_\tau))(\mathbf{O}_\tau^{\mathbf{B}_{\tau-1}},\mathbf{B}_{\tau-1})=-1\right)\right.\\
&\quad\sum_{[a_{\tau+1},\ldots,a_T]\in\{-1,1\}^{T-\tau}}\prod_{t=\tau+1}^{T} I\left(\pi_t(\widehat{f}^n_{\phi t})(\mathbf{O}_t^{[\mathbf{B}_{\tau-1},-1,\mathbf{a}_{\tau+1,t-1}]},[\mathbf{B}_{\tau-1},-1,\mathbf{a}_{\tau+1,t-1}])=a_t\right)\times\\
&\sum_{t=\tau}^{T} r\left(\mathbf{O}_t^{[\mathbf{B}_{\tau-1},-1,\mathbf{a}_{\tau+1,t-1}]},[\mathbf{B}_{\tau-1},-1,\mathbf{a}_{\tau+1,t}],O_{t+1}^{[\mathbf{B}_{\tau-1},-1,\mathbf{a}_{\tau+1,t}]}\right)+\\
&I\left(\pi_\tau(f_\tau)(\mathbf{O}_\tau^{\mathbf{B}_{\tau-1}},\mathbf{B}_{\tau-1})=1\right)\\
&\quad\sum_{[a_{\tau+1},\ldots,a_T]\in\{-1,1\}^{T-\tau}}\prod_{t=\tau+1}^{T} I\left(\pi_t(\widehat{f}^n_{\phi t})(\mathbf{O}_t^{[\mathbf{B}_{\tau-1},1,\mathbf{a}_{\tau+1,t-1}]},[\mathbf{B}_{\tau-1},1,\mathbf{a}_{\tau+1,t-1}])=a_t\right)\times\\
&\left.\left.\sum_{t=\tau}^{T} r\left(\mathbf{O}_t^{[\mathbf{B}_{\tau-1},1,\mathbf{a}_{\tau+1,t-1}]},[\mathbf{B}_{\tau-1},1,\mathbf{a}_{\tau+1,t}],O_{t+1}^{[\mathbf{B}_{\tau-1},1,\mathbf{a}_{\tau+1,t}]}\right)\right\}.\right.
\end{aligned}$$

Let $F_\tau(f_{\tau+1},\ldots,f_T,G,B_1,\ldots,B_{\tau-1})$ be

$$\max_{a_\tau\in\{-1,1\}}\Bigg\{ \sum_{[a_{\tau+1},\ldots,a_T]\in\{-1,1\}^{T-\tau}}\prod_{t=\tau+1}^{T} I\left(\pi_t(f_t)(\mathbf{O}_t^{[\mathbf{B}_{\tau-1},a_\tau,\mathbf{a}_{\tau+1,t-1}]},[\mathbf{B}_{\tau-1},a_\tau,\mathbf{a}_{\tau+1,t-1}])=a_t\right) \times \sum_{t=\tau}^{T} r\left(\mathbf{O}_t^{[\mathbf{B}_{\tau-1},a_\tau,\mathbf{a}_{\tau+1,t-1}]},[\mathbf{B}_{\tau-1},a_\tau,\mathbf{a}_{\tau+1,t}],O_{t+1}^{[\mathbf{B}_{\tau-1},a_\tau,\mathbf{a}_{\tau+1,t}]}\right)\Bigg\}.$$

Let $G_\tau(f_{\tau+1},\ldots,f_T,G,B_1,\ldots,B_{\tau-1})$ be

$$\Bigg|\sum_{[a_{\tau+1},\ldots,a_T]\in\{-1,1\}^{T-\tau}}\prod_{t=\tau+1}^{T} I\left(\pi_t(f_t)(\mathbf{O}_t^{[\mathbf{B}_{\tau-1},-1,\mathbf{a}_{\tau+1,t-1}]},[\mathbf{B}_{\tau-1},-1,\mathbf{a}_{\tau+1,t-1}])=a_t\right)\times \sum_{t=\tau}^{T} r\left(\mathbf{O}_t^{[\mathbf{B}_{\tau-1},-1,\mathbf{a}_{\tau+1,t-1}]},[\mathbf{B}_{\tau-1},-1,\mathbf{a}_{\tau+1,t}],O_{t+1}^{[\mathbf{B}_{\tau-1},-1,\mathbf{a}_{\tau+1,t}]}\right) - \sum_{[a_{\tau+1},\ldots,a_T]\in\{-1,1\}^{T-\tau}}\prod_{t=\tau+1}^{T} I\left(\pi_t(f_t)(\mathbf{O}_t^{[\mathbf{B}_{\tau-1},1,\mathbf{a}_{\tau+1,t-1}]},[\mathbf{B}_{\tau-1},1,\mathbf{a}_{\tau+1,t-1}])=a_t\right)\times \sum_{t=\tau}^{T} r\left(\mathbf{O}_t^{[\mathbf{B}_{\tau-1},1,\mathbf{a}_{\tau+1,t-1}]},[\mathbf{B}_{\tau-1},1,\mathbf{a}_{\tau+1,t}],O_{t+1}^{[\mathbf{B}_{\tau-1},1,\mathbf{a}_{\tau+1,t}]}\right)\Bigg|.$$

And let $H_\tau(f_{\tau+1},\ldots,f_T,G,B_1,\ldots,B_{\tau-1})\in\{-1,1\}$ be

$$\arg\max_{a_\tau\in\{-1,1\}}\Bigg\{ \sum_{[a_{\tau+1},\ldots,a_T]\in\{-1,1\}^{T-\tau}}\prod_{t=\tau+1}^{T} I\left(\pi_t(\widehat{f}_t^{\,n})(\mathbf{O}_t^{[\mathbf{B}_{\tau-1},a_\tau,\mathbf{a}_{\tau+1,t-1}]},[\mathbf{B}_{\tau-1},a_\tau,\mathbf{a}_{\tau+1,t-1}])=a_t\right)\times \sum_{t=\tau}^{T} r\left(\mathbf{O}_t^{[\mathbf{B}_{\tau-1},a_\tau,\mathbf{a}_{\tau+1,t-1}]},[\mathbf{B}_{\tau-1},a_\tau,\mathbf{a}_{\tau+1,t}],O_{t+1}^{[\mathbf{B}_{\tau-1},a_\tau,\mathbf{a}_{\tau+1,t}]}\right)\Bigg\}.$$

Using this shorthand notation, we have

$$\begin{aligned}
&\mathrm{E}\left\{\gamma_{\pi_{q_0,\ldots,\tau-1},\pi_\tau(f_\tau),\pi_{t+1}(\widehat{f}^n_{\phi\tau+1}),\ldots,\pi_T(\widehat{f}^n_{\phi T})}(\mathbf{O}_{T+1},\mathbf{B}_T)\right\} = \\
&\mathrm{E}\left\{F_\tau(\widehat{f}^n_{\phi\tau+1},\ldots,\widehat{f}^n_{\phi T},G,B_1,\ldots,B_{\tau-1})\right\} - \\
&\mathrm{E}\left\{G_\tau(\widehat{f}^n_{\phi\tau+1},\ldots,\widehat{f}^n_{\phi T},G,B_1,\ldots,B_{\tau-1})\times\right. \\
&\left. I\left(\pi_\tau(f_\tau)(\mathbf{O}_\tau^{\mathbf{B}_{\tau-1}},\mathbf{B}_{\tau-1}) \neq H_\tau(\widehat{f}^n_{\phi\tau+1},\ldots,\widehat{f}^n_{\phi T},G,B_1,\ldots,B_{\tau-1})\right)\right\}.
\end{aligned}$$

Introducing the surrogate ϕ we let $\widehat{f}^n_{\phi\tau}$ be

$$\begin{aligned}
\widehat{f}^n_{\phi\tau} = \arg\min_{f_\tau\in\mathcal{F}_\tau} \mathrm{E}_n &\left\{G_\tau(\widehat{f}^n_{\phi\tau+1},\ldots,\widehat{f}^n_{\phi T},G,B_1,\ldots,B_{\tau-1})\times\right. \\
&\left.\phi\left(f_\tau(\mathbf{O}_\tau^{\mathbf{B}_{\tau-1}},\mathbf{B}_{\tau-1})H_\tau(\widehat{f}^n_{\phi\tau+1},\ldots,\widehat{f}^n_{\phi T},G,B_1,\ldots,B_{\tau-1})\right)\right\}.
\end{aligned}$$

Hence, we obtain the policy $\pi(\widehat{f}^n_\phi) = (\pi_0(\widehat{f}^n_{\phi 0}),\ldots,\pi_T(\widehat{f}^n_{\phi T}))$, by solving $T+1$ weighted classification problems with a surrogate ϕ. Next we a derive finite sample upper bound on

$$V(\pi^*) - V(\pi(\widehat{f}^n_\phi)).$$

Start from Lemma 9:

$$
\begin{aligned}
&V(\pi^*) - V(\pi(\widehat{f}^n_\phi)) \leq \\
&\sum_{\tau=0}^{T} L^\tau \left[\mathrm{E}\left\{ \gamma_{\pi_{q_{0,\tau-1}},\pi^*_\tau,\pi_{\tau+1}(\widehat{f}^n_{\phi\tau+1}),\ldots,\pi_T(\widehat{f}^n_{\phi T})}(\mathbf{O}_{T+1},\mathbf{B}_T) \right\} - \right. \\
&\left. \mathrm{E}\left\{ \gamma_{\pi_{q_{0,\tau-1}},\pi_\tau(\widehat{f}^n_{\phi\tau+1}),\ldots,\pi_T(\widehat{f}^n_{\phi T})}(\mathbf{O}_{T+1},\mathbf{B}_T) \right\} \right] \leq \\
&\sum_{\tau=0}^{T} L^\tau \left[\sup_{f_\tau} \mathrm{E}\left\{ \gamma_{\pi_{q_{0,\tau-1}},\pi_\tau(f_\tau),\pi_{\tau+1}(\widehat{f}^n_{\phi\tau+1}),\ldots,\pi_T(\widehat{f}^n_{\phi T})}(\mathbf{O}_{T+1},\mathbf{B}_T) \right\} - \right. \\
&\left. \mathrm{E}\left\{ \gamma_{\pi_{q_{0,\tau-1}},\pi_\tau(\widehat{f}^n_{\phi\tau+1}),\ldots,\pi_T(\widehat{f}^n_{\phi T})}(\mathbf{O}_{T+1},\mathbf{B}_T) \right\} \right] = \\
&\sum_{\tau=0}^{T} L^\tau (T-\tau+1) \left[\sup_{f_\tau} \mathrm{E}\left\{ \frac{\gamma_{\pi_{q_{0,\tau-1}},\pi_\tau(f_\tau),\pi_{\tau+1}(\widehat{f}^n_{\phi\tau+1}),\ldots,\pi_T(\widehat{f}^n_{\phi T})}(\mathbf{O}_{T+1},\mathbf{B}_T)}{T-\tau+1} \right\} - \right. \\
&\left. \mathrm{E}\left\{ \frac{\gamma_{\pi_{q_{0,\tau-1}},\pi_\tau(\widehat{f}^n_{\phi\tau+1}),\ldots,\pi_T(\widehat{f}^n_{\phi T})}(\mathbf{O}_{T+1},\mathbf{B}_T)}{T-\tau+1} \right\} \right] \leq \\
&\sum_{\tau=0}^{T} L^\tau (T-\tau+1)\psi^{-1} \left[\mathrm{E}\left\{ \frac{G_\tau(\widehat{f}^n_{\phi\tau+1},\ldots,\widehat{f}^n_{\phi T},G,B_1,\ldots,B_{\tau-1})}{T-\tau+1} \times \right. \right. \\
&\left. \left. \phi\left(\widehat{f}^n_{\phi\tau}(\mathbf{O}^{\mathbf{B}_{\tau-1}}_\tau,\mathbf{B}_{\tau-1}) H_\tau(\widehat{f}^n_{\phi\tau+1},\ldots,\widehat{f}^n_{\phi T},G,B_1,\ldots,B_{\tau-1}) \right) \right\} - \right. \\
&\left. \inf_{f_\tau} \mathrm{E}\left\{ \frac{G_\tau(\widehat{f}^n_{\phi\tau+1},\ldots,\widehat{f}^n_{\phi T},G,B_1,\ldots,B_{\tau-1})}{T-\tau+1} \times \right. \right. \\
&\left. \left. \phi\left(f_\tau(\mathbf{O}^{\mathbf{B}_{\tau-1}}_\tau,\mathbf{B}_{\tau-1}) H_\tau(\widehat{f}^n_{\phi\tau+1},\ldots,\widehat{f}^n_{\phi T},G,B_1,\ldots,B_{\tau-1}) \right) \right\} \right],
\end{aligned}
$$

where in the first inequality we assume that the sup is attainable and the second inequality follows from [8] as in Sec. 4.5. As was done previously, we divide the bound into an estimation error, which will be bounded with high probability, and an

approximation error

$$\gamma_\tau(\mathcal{F}_\tau, \widehat{f}^n_{\phi\tau+1}, \ldots, \widehat{f}^n_{\phi T}) =$$
$$\inf_{f_\tau \in \mathcal{F}_\tau} \mathrm{E}\left\{ \frac{G_\tau(\widehat{f}^n_{\phi\tau+1}, \ldots, \widehat{f}^n_{\phi T}, G, B_1, \ldots, B_{\tau-1})}{T-\tau+1} \times \right.$$
$$\left. \phi\left(f_\tau(\mathbf{O}_\tau^{\mathbf{B}_{\tau-1}}, \mathbf{B}_{\tau-1}) H_\tau(\widehat{f}^n_{\phi\tau+1}, \ldots, \widehat{f}^n_{\phi T}, G, B_1, \ldots, B_{\tau-1}) \right) \right\} -$$
$$\inf_{f_\tau} \mathrm{E}\left\{ \frac{G_\tau(\widehat{f}^n_{\phi\tau+1}, \ldots, \widehat{f}^n_{\phi T}, G, B_1, \ldots, B_{\tau-1})}{T-\tau+1} \times \right.$$
$$\left. \phi\left(f_\tau(\mathbf{O}_\tau^{\mathbf{B}_{\tau-1}}, \mathbf{B}_{\tau-1}) H_\tau(\widehat{f}^n_{\phi\tau+1}, \ldots, \widehat{f}^n_{\phi T}, G, B_1, \ldots, B_{\tau-1}) \right) \right\} \Big].$$

Let $\overline{\gamma}_\tau(\mathcal{F}) = \sup_{f_{\tau+1} \in \mathcal{F}_{\tau+1}, \ldots, f_T \in \mathcal{F}_T} \gamma_\tau(\mathcal{F}_\tau, f_{\tau+1}, \ldots, f_T)$. The estimation error can be bounded by distances from ensemble means to empirical means:

$$\mathrm{E}\left\{ \frac{G_\tau(\widehat{f}^n_{\phi\tau+1}, \ldots, \widehat{f}^n_{\phi T}, G, B_1, \ldots, B_{\tau-1})}{T-\tau+1} \times \right.$$
$$\left. \phi\left(\widehat{f}^n_{\phi\tau}(\mathbf{O}_\tau^{\mathbf{B}_{\tau-1}}, \mathbf{B}_{\tau-1}) H_\tau(\widehat{f}^n_{\phi\tau+1}, \ldots, \widehat{f}^n_{\phi T}, G, B_1, \ldots, B_{\tau-1}) \right) \right\} -$$
$$\inf_{f_\tau \in \mathcal{F}_\tau} \mathrm{E}\left\{ \frac{G_\tau(\widehat{f}^n_{\phi\tau+1}, \ldots, \widehat{f}^n_{\phi T}, G, B_1, \ldots, B_{\tau-1})}{T-\tau+1} \times \right.$$
$$\left. \phi\left(f_\tau(\mathbf{O}_\tau^{\mathbf{B}_{\tau-1}}, \mathbf{B}_{\tau-1}) H_\tau(\widehat{f}^n_{\phi\tau+1}, \ldots, \widehat{f}^n_{\phi T}, G, B_1, \ldots, B_{\tau-1}) \right) \right\} \leq$$
$$2 \sup_{f_\tau \in \mathcal{F}_\tau, \ldots, f_T \in \mathcal{F}_T} \Big| \mathrm{E}\left\{ \frac{G_\tau(f_{\tau+1}, \ldots, f_T, G, B_1, \ldots, B_{\tau-1})}{T-\tau+1} \times \right.$$
$$\left. \phi\left(f_\tau(\mathbf{O}_\tau^{\mathbf{B}_{\tau-1}}, \mathbf{B}_{\tau-1}) H_\tau(f_{\tau+1}, \ldots, f_T, G, B_1, \ldots, B_{\tau-1}) \right) \right\} -$$
$$\mathrm{E}_n\left\{ \frac{G_\tau(f_{\tau+1}, \ldots, f_T, G, B_1, \ldots, B_{\tau-1})}{T-\tau+1} \times \right.$$
$$\left. \phi\left(f_\tau(\mathbf{O}_\tau^{\mathbf{B}_{\tau-1}}, \mathbf{B}_{\tau-1}) H_\tau(f_{\tau+1}, \ldots, f_T, G, B_1, \ldots, B_{\tau-1}) \right) \right\} \Big|.$$

Next, we bound the P-dimension of the associated function class to obtain a finite sample upper bound on this term. Let $\mathcal{H} = \{H_\tau : f_\tau \in \mathcal{F}_\tau, \ldots, f_T \in \mathcal{F}_T\}$. Let S be a set of $2n$ realizations of G and $B_1, \ldots, B_{\tau-1}$. We now compute an upper bound on $|\mathcal{H}|_S|$. The question is how many distinct collections of $2n$ pairs of paths

following the two possible actions at stage τ can the classes $\Pi_{\tau+1}(\mathcal{F}_{\tau+1}), \ldots, \Pi_T(\mathcal{F}_T)$ realize on a set of $2n$ trajectory trees. Each such two collections of $2n$ paths (one collection following action -1 and one following action 1) will result in a collection of two cumulative rewards which then will be compared pair by pair on every tree to generate an element in $|\mathcal{H}|_S|$. By Sauer's lemma [119], on the collection of histories $\{\mathbf{O}_{\tau+1}^{[\mathbf{B}_{\tau-1},-1]i}, [\mathbf{B}_{\tau-1},-1]^i\}_{i=1}^{2n}$ the policy class $\Pi_{\tau+1}(\mathcal{F}_{\tau+1})$ can realize at most $\left(\frac{2n\mathrm{e}}{d_{\tau+1}}\right)^{d_{\tau+1}}$ distinct policies. Each of these policies is generating a realization of the $2n$ next histories, on which the policy class $\pi_{\tau+2}(\mathcal{F}_{\tau+2})$ can realize at most $\left(\frac{2n\mathrm{e}}{d_{\tau+2}}\right)^{d_{\tau+2}}$ distinct policies. Continuing until the policy class $\pi_T(\mathcal{F}_T)$, we obtain that on the collection of histories $\{\mathbf{O}_{\tau+1}^{[\mathbf{B}_{\tau-1},-1]i}, [\mathbf{B}_{\tau-1},-1]^i\}_{i=1}^{2n}$, the composition of the decision rules classes can realize at most

$$\prod_{t=\tau+1}^{T} \left(\frac{2n\mathrm{e}}{d_t}\right)^{d_t}$$

distinct policies. Another set of

$$\prod_{t=\tau+1}^{T} \left(\frac{2n\mathrm{e}}{d_t}\right)^{d_t}$$

distinct policies can be realized on the set $\{\mathbf{O}_{\tau+1}^{[\mathbf{B}_{\tau-1},1]i}, [\mathbf{B}_{\tau-1},1]^i\}_{i=1}^{2n}$. Each of the elements of these two sets define a set of cumulative rewards on the trees. Hence the comparison of the two rewards on the paths defined by the two collections of distinct policies can have at most

$$\left[\prod_{t=\tau+1}^{T} \left(\frac{2n\mathrm{e}}{d_t}\right)^{d_t}\right]^2 = \prod_{t=\tau+1}^{T} \left(\frac{2n\mathrm{e}}{d_t}\right)^{2d_t}$$

distinct $2n$-vectors when realized on the $2n$ trajectory trees.

Let $\mathcal{G}_\tau = \{G_\tau : f_\tau \in \mathcal{F}_\tau, \ldots, f_T \in \mathcal{F}_T\}$. As shown above, each element of the

difference in the definition of G_τ can realize at most

$$\prod_{t=\tau+1}^{T} \left(\frac{2n\mathrm{e}}{d_t}\right)^{d_t}$$

distinct $2n$-vectors on the set of histories. Therefore $|G|_S|$ is bounded as well by

$$\prod_{t=\tau+1}^{T} \left(\frac{2n\mathrm{e}}{d_t}\right)^{2d_t}.$$

Now let $\mathcal{L} = \{G_\tau \phi(f_\tau H_\tau) : f_\tau \in \mathcal{F}_\tau, \ldots, f_T \in \mathcal{F}_T\}$. Suppose $\{v^1, v^2, \ldots, v^k\}$ is an external ϵ/μ-cover for $\mathcal{F}_\tau|_S$. For every v^j construct $\prod_{t=\tau+1}^{T} \left(\frac{2n\mathrm{e}}{d_t}\right)^{4d_t}$ vectors that take all possible values of $G_\tau \phi(v^j H_\tau)$ (there is a slight abuse of notation, since G_τ and H_τ, realized on the data, as well as v^j are $2n$-vectors). Each one of these $2n$-vectors has entrees

$$G_\tau(f_{\tau+1}, \ldots, f_T, G^i, B_1^i, \ldots, B_{\tau-1}^i)\phi\left(v_i^j H_\tau(f_{\tau+1}, \ldots, f_T, G^i, B_1^i, \ldots, B_{\tau-1}^i)\right),$$
$$i = 1, \ldots, 2n$$

for some set of functions $f_{\tau+1}, \ldots, f_T$. Now consider a sequence $f_\tau, \ldots, f_T$, which define $G_\tau \phi(f_\tau H_\tau)$ an element of $\mathcal{L}$. Then there must exist a vector v^j, for which

$$\frac{1}{2n} \sum_{i=1}^{n} |f_\tau(G^i, B_1^i, \ldots, B_{\tau-1}^i) - v_i^j| < \epsilon/\mu.$$

Assume that ϕ satisfies the Lipschitz condition (4.5.18). Then, multiplying this vector, element-wise, with H_τ of $f_\tau, \ldots, f_T$ realized on the trajectory trees, taking the function ϕ, and multiplying with G_τ of $f_\tau, \ldots, f_T$, again, realized on the trajectory trees is an external ϵ-cover of $\mathcal{L}|_S$ by the Lipschitz condition of ϕ and the argument

in the proof of Theorem 3. Hence, we have

$$\begin{aligned} L(\epsilon, \mathcal{L}|_S, ||\cdot||_{a1}) \leq & \prod_{t=\tau+1}^{T} \left(\frac{2n\mathrm{e}}{d_t}\right)^{4d_t} L(\epsilon/\mu, \mathcal{F}_\tau|_S, ||\cdot||_{a1}) \\ \leq & 2 \prod_{t=\tau+1}^{T} \left(\frac{2n\mathrm{e}}{d_t}\right)^{4d_t} \left(\frac{2\mathrm{e}\mu}{\epsilon} \ln \frac{2\mathrm{e}\mu}{\epsilon}\right)^{d_\tau} \end{aligned}$$

Given this bound [119, Theorem 7.5] asserts that the probability, over n trajectory trees, that

$$\begin{aligned} &\sup_{f_\tau \in \mathcal{F}_\tau, \ldots, f_T \in \mathcal{F}_\mathcal{T}} \Bigg| \mathrm{E}\left\{ \frac{G_\tau(f_{\tau+1}, \ldots, f_T, G, B_1, \ldots, B_{\tau-1})}{T-\tau+1} \times \right. \\ &\left. \phi\left(f_\tau(\mathbf{O}_\tau^{\mathbf{B}_{\tau-1}}, \mathbf{B}_{\tau-1}) H_\tau(f_{\tau+1}, \ldots, f_T, G, B_1, \ldots, B_{\tau-1})\right)\right\} - \\ &\mathrm{E}_n\left\{ \frac{G_\tau(f_{\tau+1}, \ldots, f_T, G, B_1, \ldots, B_{\tau-1})}{T-\tau+1} \times \right. \\ &\left. \phi\left(f_\tau(\mathbf{O}_\tau^{\mathbf{B}_{\tau-1}}, \mathbf{B}_{\tau-1}) H_\tau(f_{\tau+1}, \ldots, f_T, G, B_1, \ldots, B_{\tau-1})\right)\right\} \Bigg| > \epsilon \end{aligned}$$

is less then or equal to

$$4 \prod_{t=\tau+1}^{T} \left(\frac{2n\mathrm{e}}{d_t}\right)^{4d_t} \left(\frac{16\mathrm{e}\mu}{\epsilon} \ln \frac{16\mathrm{e}\mu}{\epsilon}\right)^{d_\tau} \exp(-n\epsilon^2/32).$$

By the union bound, the probability, over n trajectory trees, that

$$\begin{aligned} &\bigcup_{\tau=0}^{T} \sup_{f_\tau \in \mathcal{F}_\tau, \ldots, f_T \in \mathcal{F}_\mathcal{T}} \Bigg| \mathrm{E}\left\{ \frac{G_\tau(f_{\tau+1}, \ldots, f_T, G, B_1, \ldots, B_{\tau-1})}{T-\tau+1} \times \right. \\ &\left. \phi\left(f_\tau(\mathbf{O}_\tau^{\mathbf{B}_{\tau-1}}, \mathbf{B}_{\tau-1}) H_\tau(f_{\tau+1}, \ldots, f_T, G, B_1, \ldots, B_{\tau-1})\right)\right\} - \\ &\mathrm{E}\left\{ \frac{G_\tau(f_{\tau+1}, \ldots, f_T, G, B_1, \ldots, B_{\tau-1})}{T-\tau+1} \times \right. \\ &\left. \phi\left(f_\tau(\mathbf{O}_\tau^{\mathbf{B}_{\tau-1}}, \mathbf{B}_{\tau-1}) H_\tau(f_{\tau+1}, \ldots, f_T, G, B_1, \ldots, B_{\tau-1})\right)\right\} \Bigg| > \epsilon \end{aligned}$$

is less than or equal to

$$\sum_{\tau=0}^{T} 4 \prod_{t=\tau+1}^{T} \left(\frac{2n\mathrm{e}}{d_t}\right)^{4d_t} \left(\frac{16\mathrm{e}\mu}{\epsilon} \ln \frac{16\mathrm{e}\mu}{\epsilon}\right)^{d_\tau} \exp(-n\epsilon^2/32).$$

This leads to the following result

Theorem 5. *With probability greater than $1-\delta$ over a set n of trajectory trees,*

$$V(\pi^*) - V(\pi(\widehat{f}_\phi^n)) \leq \sum_{\tau=0}^{T} 2^\tau (T-\tau+1) \psi^{-1}(2\epsilon + \overline{\gamma}_\tau(\mathcal{F}))$$

for n satisfying

$$\sum_{\tau=0}^{T} 4 \prod_{t=\tau+1}^{T} \left(\frac{2n\mathrm{e}}{d_t}\right)^{4d_t} \left(\frac{16\mathrm{e}\mu}{\epsilon} \ln \frac{16\mathrm{e}\mu}{\epsilon}\right)^{d_\tau} \exp(-n\epsilon^2/32) < \delta.$$

4.7 Concluding Remarks

Theorems 4 and 5 make no assumptions on the underlying distribution and are in this sense worst case bounds. The drawback of deriving these worst case bounds is that they are usually too loss to be of practical use. By imposing regularity conditions on the underlying distribution, one can obtain faster uniform convergence rates of the empirical means to their expectation (see e.g. [8, Sec. 3] and references therein), which then lead to faster convergence rates of the average value functions. Another handicap of the given bounds is their dependency on the complexity measures of the approximation classes. For example, the bound [119, p. 412] on the P-dimension of the feed forward neural network that is reported in the next chapter, is 1.78×10^6. When plugging this value in Theorem 5, for the $T = 4$ problem in the next chapter, one obtains that for the case $\epsilon = \delta = 0.01$, n has to be of the order of 10^{14} for the bound to hold. In practice we obtained good results with $n = 10,000$ samples.

Hence, to apply the Theorems 4 and 5 to the type of problems described in the next chapter, tighter bounds on the VC-dimension of neural networks must be derived. This is beyond the scope of this thesis.

Motivated by the reduction in [62] we focused on the binary action space case. The approximate dynamic programming algorithm, however, can be applied directly, without the requirement to first reduce the problem to a binary action problem. In [25], we provide a reduction of a single-stage reinforcement learning problem to weighted classification for an arbitrary number of actions. Multi-class weighted classification problems can be solved by applying weights-sensitive classifiers or by further reducing the weighted classification problem to a standard classification problem using re-sampling methods (see [71], [1], and references therein for a description of both approaches). Theorems 2 and 4 can be easily adapted to the multiple actions case, while adapting Theorems 3 and 5 will require tailoring to the specific method used to solve the multi-class weighted classification problems.

The algorithms in [62] and [72] require the construction of the entire trajectory tree. This requires that an exponential, in the horizon T, number of calls are made to the random observation generator. It is possible to show that our algorithm requires only a polynomial number of calls. Hence, our approximate dynamic programming algorithm provides an additional saving compared to the available methods. We note that [62] discusses ways to avoid constructing the entire tree. However, in the worst case scenario, the entire tree construction is unavoidable.

Finally, we note that, while not investigated here, the algorithm and part of its analysis apply to the case in which the data set is a collection of random trajectories of the decision process (see [62] and [84]) rather than full trajectory trees.

CHAPTER 5

Optimal Sensor Scheduling via Classification Reduction of Policy Search

5.1 Introduction

The advent of agile sensing systems that collect data through a variety of sensing modalities has brought about new and exciting challenges to the field of signal processing. Agile, multi-modal, sensing (see e.g. [68] and [60]) exploit the capability of controlling the data collection process. Examples of agile sensing systems include a radar that can control its beam direction, a land mine detector that can deploy radar or seismic sensors, or a LANDSAT satellite that can control the frequency band of its radar. The key element that differentiates agile sensing systems from other data collection systems is a resource allocation constraint that precludes using all sensor modalities at all times. We formulate agile sensing as an optimization in which the system must automatically select the best sensing modality based on past observations to maximize a given objective function while minimizing the data collection cost.

When formulated as a sequential choice[1] of experiments problem [37], the agile

[1]The key difference from the related sequential design of experiment problem is that instead

sensing problem consists of an episodic task that is divided into a sequence of decision epochs. Each episode begins as the first observation is collected. Then, at each subsequent decision epoch two decisions are made. The first one is to decide if the amount of information collected thus far is sufficient for making inference (detection or estimation) on the data with a desired accuracy or whether more observations are required. This first decision also determines the choices available at the second decision. If more observations are required, the next best sensor modality needs to be determined. If the information is deemed sufficient for inference, the final estimation or detection decision is made. Every sensor modality has an associated deployment cost and a decision rule must balance the expected information gain from a sensor deployment, which results in improved inference capabilities, with the deployment cost. The collection of decision rules, i.e., the sequence of mappings from past observations to the decision space, is called a policy and the goal is to find a policy that optimally trades-offs the overall average sensor deployment costs and the estimation or detection performance, e.g., mean squared estimation error or classification error rate.

The problem of finding optimal policies for sequential choice of experiments suffers from the curse-of-dimensionality [11] and scenarios in which a closed form solution for the optimal policy exists are rare. Past research has focused on the asymptotic regime in which one assumes a large number of data collection iterations (or sensor dwells) and low sensor deployment cost (see [63] and references therein). Another focus has been on "experiment sufficiency" – when is one experiment (or sensor modality) always better than another experiment (see [50] and references therein).

Here, we take a different approach. We assume that the underlying model is unknown and aim at finding *approximate* solutions to the optimal policy. In particular,

of adapting a set of continuous experiment parameters, here we choose from a finite set of fixed experiments.

in the absence of a model, optimal policies are approximated from data using a generative model, where data is generated by a simulator or collected in a field experiment. It is shown that this problem formulation falls into the class of reinforcement learning problems and the Classification Reduction of Policy Search (CROPS) methodology of the previous chapter is applied. Two case studies are reported as well. The first is the problem of finding sensor scheduling policies for land-mine detection. For this problem a simulator is used to generate data which is then used for policy search. The second problem is to perform optimal waveform selection for a multi-band radar on a land classification satellite. In this application competitive policies are found from experimental LANDSAT data.

5.2 Problem Formulation

Let $X_1 \in \mathcal{X}_1, X_2 \in \mathcal{X}_2, \ldots, X_K \in \mathcal{X}_K$ be K random variables that correspond to the outputs of K sensors or K sensor modalities. Note that each of these random variables lies, in general, in a different space. We append each random variable with its index so that a value of an observation also indicates which sensor was used to collect it. Let $Y \in \mathcal{Y}$ be a discrete random variable that represent the state of nature whose value we try to predict. The presented results can also be applied when Y is a continuous random variable, whose value we try to estimate, but we focus on the detection problem for concreteness.

A policy π specifies which sensor to deploy first, say sensor k. Then, based of the value of X_k, the policy determines if an accurate prediction of Y is possible, and if so, what is the best prediction, or, otherwise, which is the next best sensor to deploy to collect additional data. This process continues until either a prediction of Y is made or all available sensors are deployed. We assume that each sensor can

be applied at most once and hence, the total number observations is bounded by K. Therefore, a policy π is sequence of $K+1$ decision rules $\pi = [\pi_1, \pi_2, \ldots, \pi_{K+1}]$. This assumption is valid when the randomness in the process, e.g. the observation noise, is governed by clutter that cannot be averaged out by repeated measurements, rather than by thermal noise. Note that π_1 simply indexes the first sensor to deploy (excluding the possibility of predicting Y without taking any observations), and hence, $\pi_1 \in \{1, 2, \ldots, K\}$. Also note that π_{K+1} is used only if at all the decision epochs the decision was to defer the prediction of Y and deploy another sensor. The decision rule π_{K+1} is a map from $\mathcal{X}_1 \times \mathcal{X}_2 \times \ldots \times \mathcal{X}_K$ to $\mathcal{Y}$. If the objective is to try to minimize the detection error, then it is well known that the optimal map is the Bayes classifier [54]

$$\pi^*_{K+1}(x_1, x_2, \ldots, x_K) = \arg\max_{y \in \mathcal{Y}} \Pr\{Y = y | X_1 = x_1, X_2 = x_2, \ldots, X_K = x_K\}.$$

The domain and range of the decision rules for stages $2, \ldots, K$ depend on the sequence of sensors deployed up to the decision time. For example, if $\pi_1 = k$, then

$$\pi_2 : \mathcal{X}_k \rightarrow (\{1, 2, \ldots, K\} \setminus k) \bigcup \mathcal{Y}.$$

If $\pi_2(x_k) \in (\{1, 2, \ldots, K\} \setminus k)$ then the decision is to take another observation using sensor $\pi_2(x_k)$. Alternatively, if $\pi_2(x_k) \in \mathcal{Y}$, then the decision is that the amount of information is sufficient and $\pi_2(x_k)$ is the predictor of Y. Instead of explicitly defining the policy through a sequence of mappings whose domains and ranges depend on past decisions and observations, we let $Z = [X_1, \ldots, X_K]$ and define the policy π as a two-dimensional function of Z. Given, the value of Z, its first argument $[\pi(Z)]_1$ is the resulting sequence of sensors that were deployed prior to the final decision and its

second argument $[\pi(Z)]_2$ is the prediction for Y. Note that in general, only a subset of the elements of Z are observable at the time the final decision is made.

Denote by $P_c(\pi) = \Pr\{[\pi(Z)]_2 = Y\}$ the probability of correctly predicting the value of Y based on the data collected according to the policy π, by $C([\pi(Z)]_1)$ the cost associated with the sequence of sensor deployments $[\pi(Z)]_1$, e.g., the number of sensor dwells, and by $\mathrm{E}\{C([\pi(Z)]_1)\}$ the expected cost. We assume that the cost of the deployment of a sequence of sensors is the sum of the costs of deploying each of the sensors, and hence, does not depend on the order of deployment. The optimal policy π^* is the policy that maximizes

$$P_c(\pi) - \lambda \mathrm{E}\{C([\pi(Z)]_1)\}, \tag{5.2.1}$$

where λ is a tuning parameter that trades off the cost of data collection and the cost of prediction error. Under certain regularity conditions, the optimal policy can be defined though backward induction (see e.g. [93]). However, when $\mathcal{X}_1, \ldots, \mathcal{X}_K$ are continuous or discrete and large, the solution becomes intractable. Furthermore, even when $\mathcal{X}_1, \ldots, \mathcal{X}_K$ are finite and relatively small, the backward induction iterations require computing expectations with respect to the joint distribution of Z and Y.

Here, we allow $\mathcal{X}_1, \ldots, \mathcal{X}_K$ to be continuous or discrete and large, and consider the case in which the joint distribution of Z and Y is unknown. We assume that n realizations of (Z, Y) are available and the goal is find a policy that maximizes (5.2.1) based on this data set. Hence, this is a model free instance of the sequential choice of experiments problem as formulated in [37], which, to the best of our knowledge, has not been considered previously in the literature.

5.3 Stochastic Decision Process Formulation

The formulation of our sequential choice of experiments problem as a finite-horizon partially observable stochastic decision process discussed in the previous chapter consists of several elements:

- The decision epochs determine the times at which an action is executed. In the discrete model adopted here, decision epochs occur at $t = 0, \ldots, \tau$. At every decision epoch either another observation is collected, or a final prediction of Y is made. In the later case the processes terminates. Therefore, τ is a random variable that depends on the deployed policy and Z.

- The system's state is the realization of Y which is fixed throughout the episode.

- The state at time zero is a random variable with distribution D over $\mathcal{Y}$.

- The state of the system cannot be directly observed but instead after every decision epoch $t = 0, \ldots, \tau$, in which the decision is to collect another observation, a noisy observation O_t of the systems' state is collected. The domain and distribution of the observation depends on the underlying systems' state Y and the deployed sensor. Denote by $\overline{O}_t = [O_0, O_1, \ldots, O_t]$ the observations up to and including time $t < \tau$, and note that $\overline{O}_t$ is a subset of Z.

- At every decision epoch $0 \leq t \leq \tau$ the agent chooses an action a_t, based on the past observations, from a set of possible actions – the action space $\mathcal{A}_t$. Though not explicitly appearing in the notation, the set of available actions $\mathcal{A}_t$ may depend on the past actions. In our application, only actions that correspond to sensors that have not be previously deployed can be taken.

- The action of making the prediction of Y is a termination action that ends the process.

- We note that even though in our formulation the state of the system is fixed throughout the episode, the results can be generalized to the case in which upon taking action a at state y, the system makes a transition to state y' according to a transition probability $P_{y,a}$. In other wards, it is possible to generalize to the case in which the system's states evolve as a Markov process. This generalization is important for cases in which sensor deployment may be sensed by the target and lead to changes in the target's state as in [60].

- A reward $r(Y, a)$ is received after each time an action is taken. When a sensor is deployed to collect another observation, $r(Y, a)$ is minus the cost of deploying sensor a regardless of the state of the system. In the application below, the cost is the same for all sensor modalities, and it is denoted by c. When the final prediction is made a reward of one unit is received only if the prediction $a = \widehat{Y}(\overline{O}_{\tau-1})$ equals Y, i.e., $r(Y, a) = I(a = Y)$, where I is the indicator function that equals one when its argument is true and zero otherwise.

- A policy π is a sequence of decision rules, or mappings from past observations to the action spaces, which specifies the action to take at each decision epoch. The policy is composed of $K + 1$ decision rules $(\pi_0, \pi_1, \ldots, \pi_K)$, however, if the termination action is taken prior to decision epoch K then not all decision rules are executed.

A typical episode is a sequence

$$a_0 \to O_0 \to a_1(O_0) \to O_1 \to a_2(\overline{O}_1) \ldots$$
$$O_{\tau-1} \to a_\tau(\overline{O}_{\tau-1}) = \widehat{Y}(\overline{O}_{\tau-1}),$$

where a_0 is the first decision to deploy a sensor before any observations were collected, $O_0, O_1, \ldots, O_{\tau-1}$ are the observations whose domains and distributions depend on Y and the decisions $a_0, a_1, \ldots, a_{\tau-1}$, respectively, and $a_\tau(\overline{O}_{\tau-1})$ is a decision that the past observations are sufficient for making a prediction on Y, and it specifies the predictor $\widehat{Y}(\overline{O}_{\tau-1})$. The objective is to find a policy π that maximizes the expected sum of rewards:

$$V(\pi) = \mathrm{E}_\pi \left\{ \sum_{t=0}^{\tau} r(Y, \pi_t(\overline{O}_{t-1})) \right\}, \tag{5.3.2}$$

where the expectation is taken with respect to the joint distribution of Z and Y, which, through π, induce a distribution on the observations $O_0, O_1, \ldots, O_{\tau-1}$. The expected sum of rewards $V(\pi)$ is called the value of the policy π.

It is well known that when the underlying joint distribution of the system state and the observations is known and the observations can take a finite number of possible values, it is possible to formulate the problems in terms of the information state and solve for the optimal policy [59]. In our setting, however, the joint distribution is unknown and the observations are, in general, continuous random variables. Approximating the optimal policy in this case is a classic problem in reinforcement learning. Here, we adopt the generative model assumption of [62]. Under this assumption, the initial distribution D and the distribution of the observations conditioned on the system state and the deployed sensor are unknown but it is possible to generate realizations of the system state Y according to D and observations conditioned on arbitrary state Y and deployed sensor. In particular, we assume that we have n realizations of the pair (Z, Y) denoted by $\{(Z_1, Y_1), (Z_2, Y_2), \ldots, (Z_n, Y_n)\}$. Note that given a realization (Z_1, Y_1) it is possible to generate the entire decision tree associated with the sequential choice of experiment problem. Given a realization (Z_1, Y_1)

and a policy π, it is possible to follow the path that a system that uses π will follow and compute the sum of rewards for this realization. Prior to the prediction of Y, the rewards are minus the sensor deployment costs, and, at the prediction epoch, a unit reward is received only if $\widehat{Y}(\overline{O}_{\tau-1}) = Y_1$, where $\widehat{Y}(\overline{O}_{\tau-1})$ is chosen by following the path induced by π.

Now, consider a class of policies Π, i.e., each element $\pi \in \Pi$ is a sequence of decision rules $\pi = (\pi_0, \pi_1, \ldots, \pi_K)$. It is possible to estimate the value $V(\pi)$ (5.3.2) of any policy in the class from the set of trajectory trees by simply averaging the sum of rewards on each tree along the path that agrees with the policy [62]. A policy specifies the action to take at each decision epoch and so there is exactly one path in every tree that agrees with a given policy. Denote by $\widehat{V}^i(\pi)$ the observed sum of rewards on the i'th tree along the path that corresponds to the policy π. Then the value of the policy π is estimated by

$$\widehat{V}_n(\pi) = n^{-1} \sum_{i=1}^{n} \widehat{V}^i(\pi). \tag{5.3.3}$$

In [62], the authors show that with high probability (over the data set) $\widehat{V}_n(\pi)$ converges uniformly (over Π) to $V(\pi)$ with rates that depend on the VC-dimension of the policy class. This result motivates the use of policies π with high $\widehat{V}_n(\pi)$, since with high probability these policies have high values of $V(\pi)$.

In the previous chapter it is shown that while the task of finding the global optimum within a class of non-stationary policies may be overwhelming, the approximate dynamic programming algorithm leads to a sequence of single step reinforcement learning problems which can be reduced to a sequence of weighted classification problems.

If the action space is not binary, as in the problem described below, the reduction leads to a sequence of multi-class weighted classification problems. These can be solved using re-sampling methods or heuristic extensions of methods for binary weighted classification (see [1] for both approaches), as done with the application of the k-nearest neighbor algorithm to be described below. Alternatively, one can use the reduction of [62] that converts a multi-action RL problem into a binary RL problem by introducing dummy decision epochs, as done for the application of the weights-sensitive neural network in the problem below. The reduction is operated on every stage at which more than two actions are available. Note that it is possible to describe any action as the answer to at most $\lceil \log_2(L) \rceil$ 'yes or no' questions, where L is the number of actions, and $\lceil x \rceil$ is the smaller integer larger than or equal to x. Then, every stage with more than two actions is described by the decision tree associated with these binary decision epochs. Once an intermediate decision is made, it corresponds to a transition to the same state, i.e., the state does not evolve, but the action space is halved. Only when the decision is between two actions, the chosen action is executed and a state transition occurs.

5.4 Sensor Scheduling for Land-Mine Detection

This section reviews a sequential choice of experiment problem that arises in the design of unmanned land-mine detection vehicle. The vehicle carries three sensors for performing the detection: an EMI sensor, a ground penetrating radar (GPR), and an acoustic sensor. As can be seen in Figure 5.1, the sensors have different responses under different types of land-mines and clutter. In addition, deploying a sensor takes time and energy and hence not all sensors are deployed at every potential land-mine location. Upon reaching a new location, in which a land-mine is potentially present,

a policy that trades of the cost of a sensor deployment and detection probability determines the first sensor to deploy. Based on the collected measurement, either a prediction regarding the presence of the land-mine is made or a second sensor is deployed. Finally, based on the output of the first two deployed sensors, either a prediction regarding the presence of the land-mine is made or a third sensor is deployed followed by the final prediction based on all three measurements. The goal is to maximize the probability of correct detection minus a constant $c > 0$ (5.2.1) times the number of sensor dwells.

Since there are a total of three sensors $Z = [X_1, X_2, X_3]$. The state space is binary $\mathcal{Y} = \{0, 1\}$, where $Y = 0$ means no land-mine is present and $Y = 1$ indicates the presence of a land-mine. The decision tree associated with this problem is presented in Figure 5.2.

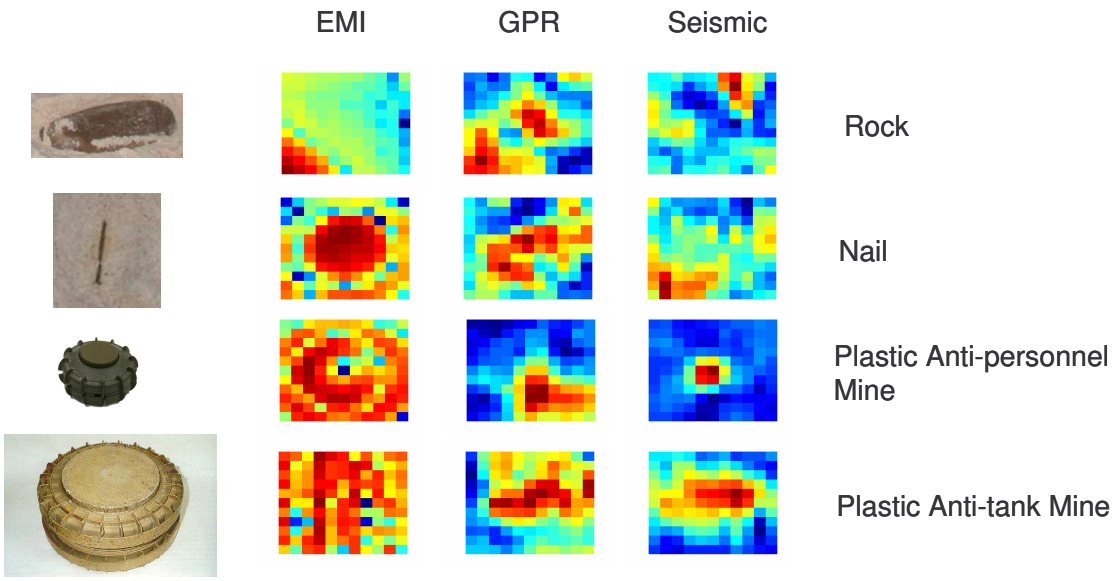

Figure 5.1: Sensors signatures for several land-mine and clutter types.

Figure 5.4 summarizes the features extracted from each sensor and their expected signatures under different scenarios. In the simulation, one of the possible eight scenarios was first chosen randomly. Then, a realization of each of the features,

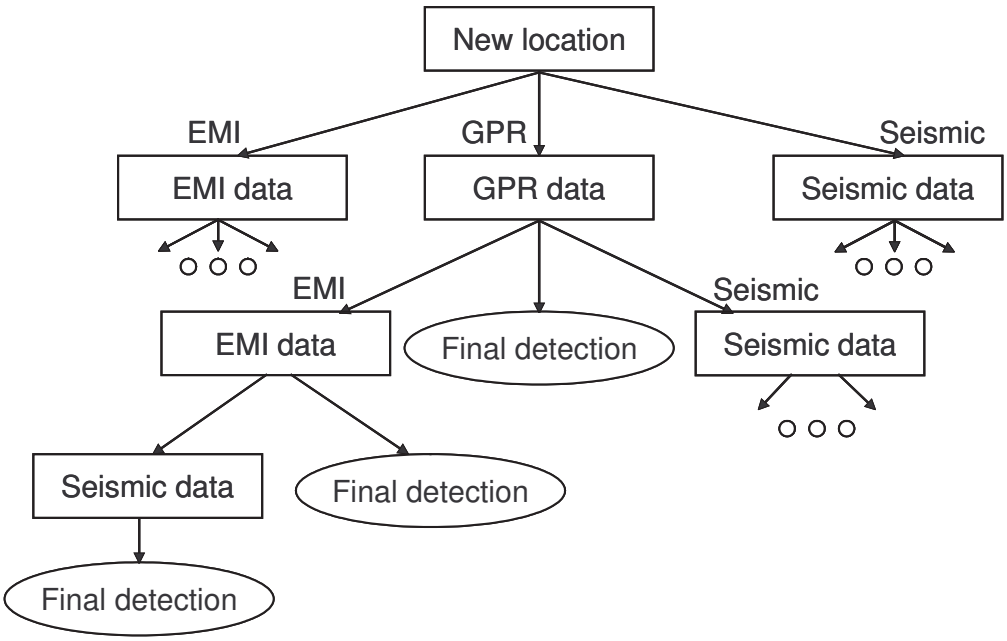

Figure 5.2: The decision tree associated with the land-mine detection problem.

which together compose Z, is generated as a Gaussian random variable with means 0, 0.5, or 1, corresponding to low, medium, or high, respectively. The covariances of sensors 1, 2, and 3, were $0.5I$, $0.45I$ and 0.1, respectively, where I is the 2-dimensional identity matrix. These values of means and covariances were chosen in correspondence with experiments that were conducted in a sand box [80]. Hence the marginal distribution of the vector of sensor outputs is a five-dimensional eight-component Gaussian mixture.

Before searching for the optimal sensor scheduling policy, the classifiers

$$\widehat{Y}(O_1), \widehat{Y}(\overline{O}_2), \widehat{Y}(\overline{O}_3)$$

for all possible combinations of sensor selections

$$X_1, X_2, X_3,$$
$$(X_1, X_2), (X_1, X_3), (X_2, X_3),$$
$$(X_1, X_2, X_3)$$

were found by training two-layer feed-forward neural networks, each with ten input and two output nodes, on 1000 samples of (Z, Y). By testing the performance of these classifiers on a separate test set of 1000 samples, we found that the best single sensor to use for detecting a land-mine is the EMI sensor, that the two best fixed sensors are GPR plus the Seismic, and that in this scenario the classifier which is based on the output of all three sensors has a probability of correct detection of 0.887. The search for the optimal sensor scheduling policy was conducted while these classifiers remained fixed. In other words, only decisions regarding whether or not to deploy a sensor, and which sensor to deploy next were considered. Since the classifiers remained fixed during the policy search, once a decision to make prediction is made, the reward is gained according to the classifier output, without trying to further optimize its performance.

As explained above, the optimal policy was approximated by introducing dummy decision epochs, so that all the decisions are binary. We then performed the nonlinear Gauss-Seidel decomposition into a sequence of single-stage binary reinforcement learning problems. Each subproblem was then converted to a weighted classification problem that was solved by a weights-sensitive two-layer feed-forward neural network with seven input and two output nodes.

Figure 5.3 summarizes the results. The horizonal axis is the average number of sensor dwells and the vertical is the probability of correct detection. The three solid

circles correspond to the performance of the best single sensor, best two sensors, and the performance when all three sensors are deployed, respectively. These points are connected by a solid line that corresponds to performance that can be achieved by randomly selecting one of these fixed sensor configurations. The crosses corresponds to the performance (estimated from a 1000 trail test set) obtained by the approximated optimal sensor scheduling policies. Each cross correspond to a different choice of c (5.2.1), ranging from $c = 0.2$ at the left lower corner and $c = 0$ at the outmost upper right cross. When $c = 0.2$ the price of taking more than a single measurement is too dear compared to the improvement in the probability of correct detection and the policy dictates making decision using only a single sensor. As c decreases, more and more observations are allowed. It is interesting to see that when c is zero, i.e, the sensor deployment cost is zero, the algorithm does not always deploy all three sensors, but achieves better performance than when all three sensors are always deployed. This happens since the classifiers used at the prediction stages are not the Bayes classifiers (in which more information can never worsen performance) but rather sub-optimal classifiers that were found by training neural networks.

It is encouraging that by training the neural networks we found a policy that accounts for generalization errors at the predictor level and do not collect the third observation when that observation might lead to a worse prediction. In summary, it can be seen that through sensor scheduling it is possible to achieve better classification performance with fewer average number of sensor dwells. The actual sensor sequences taken under the possible eight scenarios when the policy whose performance cross is circled is presented in Figure 5.4. It is seen that the optimal policy dictates that the first deployed sensor is the GPR sensor even though the optimal single sensor is the EMI sensor. This is not surprising since an optimal sensor scheduling optimizes the future sum of rewards rather than choosing the sensor whose stand

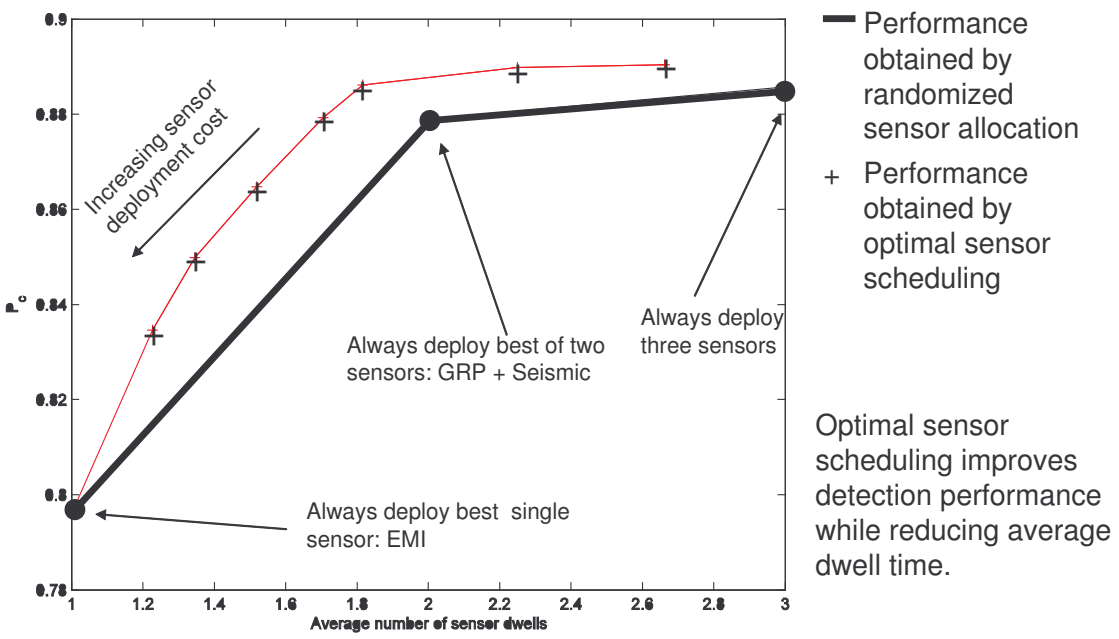

Figure 5.3: Performance of sensor-scheduling-based detection compared to detection under optimal fixed sensor allocations.

		Object Type								
		1	2	3	4	5	6	7	8	Feature
		M-AT	M-AP	P-AT	P-AP	Cltr-1	Cltr-2	Cltr-3	Bkg	Description
Sensor	EMI (1)	High	High	Medium	High	High	Low	Low	Low	Conductivity
		High	High	High	Medium	Medium	Low	Low	Low	Size
	GPR (2)	High	Medium	High	Medium	Low	Low	Low	Low	Depth
		High	Medium	High	Medum	High	High	High	Low	RCS
	Seismic (3)	High	Medium	High	Medium	Medium	Medium	Low	Low	Resonance
Optimal sequence for mean state		2 3 D	2 1 D	2 3 D	2 1 3 D	2 3 D	2 3 D	2 3 D	2 3 D	

Figure 5.4: Sensor mean responses under various scenarios. M-Metal, P-Plastic, AP-Anti personal, AT-Anti tank, Cltr-1-Hallow metal clutter, Cltr-2-Hallow non-metal clutter, Cltr-3-Non-metal non-hallow clutter, Bkg-Background.

alone performance are the best. Furthermore, only when the underlying system state is a plastic anti-personal land-mine, which has the weakest signature, does the policy dictate using all three sensors. In other cases, two sensors are sufficient for the land-mine detection.

5.5 Waveform Selection for Land Monitoring Satellite

In this section, the optimal sensor scheduling algorithm is applied to real data for the problem of waveform selection for a LANDSAT land monitoring satellite. The satellite collects a radar backscatter on a patch of land and the goal is to classify the land type based on the returned signal. Given a new probing location, the satellite can transmit one of four possible waveforms. The different waveforms correspond to different frequency bands. Therefore, $Z = [X_1, X_2, X_3, X_4]$. Each of the observations $X_1, \ldots, X_4$ is a 9-dimensional vector taking values in $[0, 255]^9$, and hence, Z is a 36-dimensional vector. There are six land types, and hence $\mathcal{Y} = \{1, 2, \ldots, 6\}$. In the public data set [109], there are 4435 points in the training set and 2000 in the test set. For a more detailed explanation of the problem see [54] chapter 13. In this section we explore, using sensor scheduling, reducing the number of waveform (frequency band) transmissions. In particular, we find policies that select the first best two frequency bands and based on the outcome determine if the remaining frequency bands are required, or whether the first two bands provide sufficient information for classifying the land type. Hence, at the first decision epoch there are six possible actions leading

to six possible measured pairs of frequency bands:

$$\{[X_1, X_2], [X_1, X_3], [X_1, X_4], ...$$
$$[X_2, X_3], [X_2, X_4], [X_3, X_4]\}.$$

The land type classifiers are the k-nearest neighbors algorithm with k set to 5, as recommended in [54] for the non-sequential problem. Two classifiers for the policy search were considered. The first is a $[7, 5, 2]$ feed-forward weights-sensitive neural network with sigmoid activation functions. The neural network is trained to minimize the objective function (ref to multi stage with surrogate objective) with ϕ being the truncated squared error loss. The second is a weights-sensitive k-nearest neighbor, where $k = 30$, chosen using leave-one-out cross validation [54]. At every given point, the k-nearest neighbor algorithm chooses the actions that minimizes the weighted classification error averaged on the point's k nearest neighbors in the training set. The performance are summarized in Figure 5.5. The crosses correspond to the performance of policies that were found by weights-sensitive k-nearest neighbor classifiers as c ranges from 0 to 0.18. The squares correspond to the performance of policies that were found by weights-sensitive $[7, 5, 2]$ feed-forward neural networks for four values of c. To study the effect of the initial network weights distribution, for each value of c, the neural networks training was initiated at four random weights selections, leading to four resulting policies. As can be seen, under both learning configurations it is possible to obtain a range of trade-offs between sensor deployment cost and classification performance. Particularly, the policy learned by the k-nearest neighbor classifier with $c = 0.02$ almost achieves the same performance as when all sensor modalities are used, but with a significant reduction in deployment cost. From comparing the performance of the k-nearest neighbor classifier based policy with the

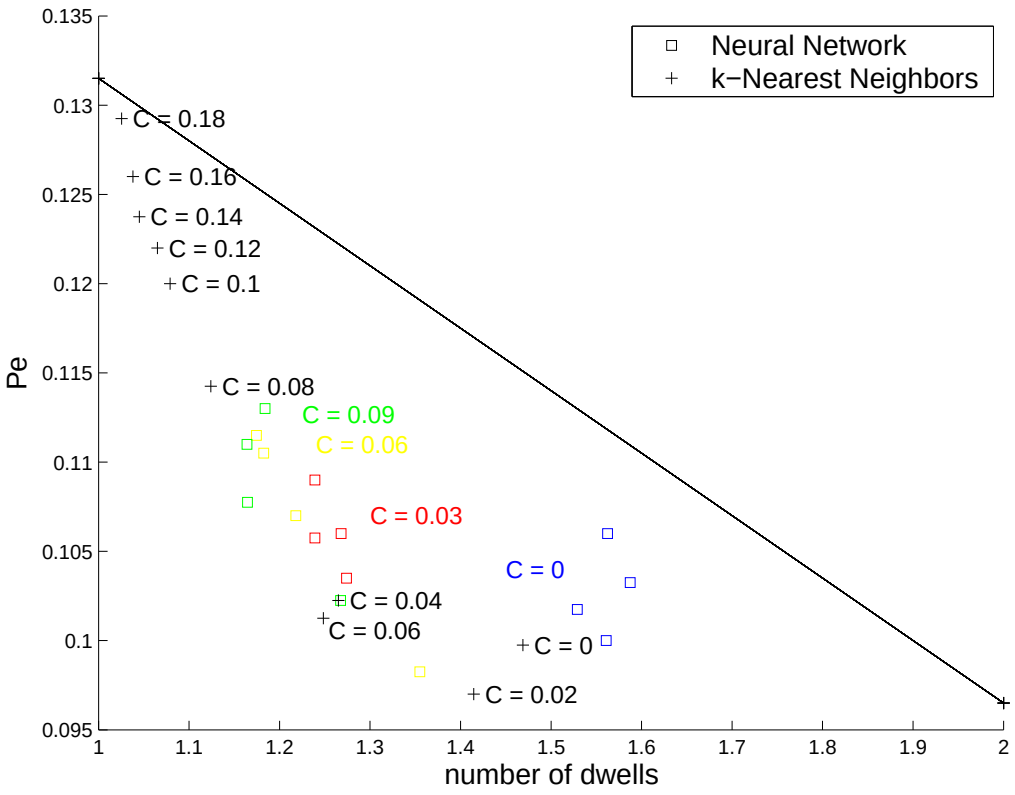

Figure 5.5: Performance of sensor scheduling algorithm for the land monitoring satellite problem.

one based on the neural networks it is seen that the performance achieved by the two architectures are comparable.

BIBLIOGRAPHY

BIBLIOGRAPHY

[1] N. Abe, B. Zadrozny, and J. Langford, "An iterative method for multi-class cost-sensitive learning," in *Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2004, pp. 3–11.

[2] S. Ahn, J. Fessler, D. Blatt, and A. Hero, "Incremental optimization transfer algorithms: application to tomography," *Submitted to: IEEE Trans. Image Process.*, 2004.

[3] S. Amari, *Differential-Geometrical Methods in Statistics*. Berlin: Springer-Verlag, 1990.

[4] C. Andrieu and A. Doucet, "Simulated annealing for maximum a posteriori parameter estimation of hidden markov models," *IEEE Trans. Inform. Theory*, vol. 46, no. 3, pp. 994 – 1004, May 2000.

[5] M. Anthony and P. L. Bartlett, *Neural Network Learning: Theoretical Foundation*. Cambridge, UK: Cambridge University Press, 1999.

[6] S. Arora, L. Babai, J. Stern, and Z. Sweedyk, "Hardness of approximate optima in lattices, codes, and linear systems," *Journal of Computer and System Sciences*, vol. 54, no. 2, pp. 317–331, 1997.

[7] J. Bagnell, S. Kakade, A. Ng, and J. Schneider, "Policy search by dynamic programming," in *Advances in Neural Information Processing Systems*, vol. 16. MIT Press, 2003.

[8] P. L. Bartlett, M. I. Jordan, and J. D. McAuliffe, "Convexity, classification, and risk bounds," *Journal of the American Statistical Association*, vol. 101, no. 473, pp. 138–156, March 2006.

[9] A. G. Barto and T. G. Dietterich, "Reinforcement learning and its relationship to supervised learning," in *Handbook of learning and approximate dynamic programming*, J. Si, A. Barto, W. Powell, and D. Wunsch, Eds. John Wiley and Sons, Inc, 2004.

[10] B. M. Bell, "The iterated kalman smoother as a Gauss-Newton method," *SIAM J. Optim.*, vol. 4, pp. 626–636, 1994.

[11] R. Bellman, *Dynamic Programming.* Princeton, NJ: Princeton University Press, 1957.

[12] S. Ben-David, N. Eiron, and P. M. Long, "On the difficulty of approximately maximizing agreements," *Journal of Computer and System Sciences*, vol. 66, no. 3, p. 496514, 2003.

[13] A. Ben-Tal, T. Margalit, and A. Nemirovski, "The ordered subsets mirror descent optimization method with applications to tomography," *SIAM J. Optim.*, vol. 12, no. 1, pp. 79–108, 2001.

[14] D. P. Bertsekas, "Incremental least squares methods and the extended Kalman filter," *SIAM J. Optim.*, vol. 6, no. 3, pp. 807–822, 1996.

[15] ——, "A new class of incremental gradient methods for least squares problems," *SIAM J. Optim.*, vol. 7, no. 4, pp. 913–926, 1997.

[16] ——, *Nonlinear programming: second edition.* Belmont, MA: Athena Scientific, 1999.

[17] D. P. Bertsekas and J. N. Tsitsiklis, *Neuro-Dynamic Programming.* Belmont, MA: Athena Scientific, 1996.

[18] ——, "Gradient convergence in gradient methods with errors," *SIAM J. Optim.*, vol. 10, no. 3, pp. 627–642, 2000.

[19] P. J. Bickel and J. A. Yahav, "On estimating the total probability of the unobserved outcomes of an experiment," in *Adaptive statistical procedures and related topics*, J. van Ryzin, Ed. Hayward, CA: Institute of Mathematical Statistics 1986.

[20] C. Biernacki, "Un test pour le maximum global de vraisemblance," *35ièmes journées de statistiques, Lyon, France SFdS'2003*, June 2003.

[21] ——, "Testing for a global maximum of the likelihood," *J. Comput. Graph. Statist.*, vol. 14, no. 3, pp. 657–674, Sept. 2005.

[22] P. Billingsley, *Probability and Measure.* New York: John Wiley and Sons, 1995.

[23] C. M. Bishop, *Neural Networks for Pattern Recognition.* Oxford, Great Britain: Oxford University Press, 1995.

[24] D. Blatt and A. Hero, "Distributed maximum likelihood for sensor networks," in *Proceedings of the 2004 IEEE International Conference on Acoustics, Speech, and Signal Processing*, Montreal, Canada, May 2004, pp. 929 – 932.

[25] D. Blatt and A. O. Hero, "From weighted classification to policy search," in *18th Annual Conference on Neural Information Processing Systems (NIPS)*, 2005.

[26] C. Bon-Sen, L. Bore-Kuen, and P. Sen-Chueh, "Maximum likelihood parameter estimation of f-arima processes using the genetic algorithm in the frequency domain," *IEEE Trans. Signal Process.*, vol. 50, no. 9, pp. 2208 – 2220, 2002.

[27] H. Bunzel, N. M. Kiefer, and T. J. Vogelsang, "Simple robust testing of hypotheses in nonlinear models," *J. Amer. Statist. Assoc.*, vol. 96, no. 455, pp. 1088–1096, Sept. 2001.

[28] C. Byrne, "Choosing parameters in block-iterative or ordered subset reconstruction algorithms," *IEEE Trans. Image Process.*, 2004, to appear.

[29] Y. Censor and G. T. Herman, "Block-iterative algorithms with underrelaxed Bregman projections," *SIAM J. Optim.*, vol. 13, no. 1, pp. 283–297, 2002.

[30] Y. Censor, A. R. D. Pierro, and M. Zaknoon, "Steered sequential projections for the inconsistent convex feasibility problem," *Nonlinear analysis: theory, methods, and application, series A*, vol. 59, pp. 385–405, 2004.

[31] J. C. Chen, K. Yao, and R. E. Hudson, "Source localization and beamforming," *IEEE Signal Processing Magazine*, vol. 19, no. 2, pp. 30–39, March 2002.

[32] M. Collins, R. E. Schapire, and Y. Singer, "Logistic regression, AdaBoost and Bregman distances," *Machine Learning*, vol. 48, no. 1-3, pp. 253–285, 2002.

[33] D. R. Cox, "Tests of separate families of hypotheses," *in Proceedings of the Fourth Berkeley Symposium on Mathematical Statistics and Probability*, vol. 1, pp. 105–123, 1961.

[34] ——, "Further results on tests of separate families of hypotheses," *Journal of the Royal Statistical Society B*, pp. 406–424, 1962.

[35] D. Crawford, "Special issue on wireless sensor networks," *Commun. ACM*, vol. 47, no. 6, June 2004.

[36] W. C. Davidon, "New least-square algorithms," *J. Optim. Theory Appl.*, vol. 18, no. 2, pp. 187–197, Feb. 1976.

[37] M. H. DeGroot, *Optimal Statistical Decisions*. New York: McGraw-Hill, 1970.

[38] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data using the em algorithm," *Ann. Roy. Statist. Soc.*, vol. 39, pp. 1–38, Dec 1977.

[39] R. E. Dorsey and W. J. Mayer, "Detection of spurious maxima through random draw tests and specification tests," *Computational Economics*, vol. 16, pp. 237–256, 2000.

[40] H. Erdogan and J. A. Fessler, "Monotonic algorithms for transmission tomography," *IEEE Tr. Med. Im.*, vol. 18, no. 9, pp. 801–14, Sept. 1999.

[41] A. H. et. al., "Highlights of statistical signal and array processing," *IEEE Signal Processing Magazine*, vol. 15, no. 5, pp. 21–64, Sept. 1998.

[42] A. Fern, S. Yoon, and R. Givan, "Approximate policy iteration with a policy language bias," in *Advances in Neural Information Processing Systems*, vol. 16, 2003.

[43] M. A. T. Figueiredo and A. K. Jain, "Unsupervised learning of finite mixture models," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 24, no. 3, pp. 381–396, Mar. 2002.

[44] S. J. Finch, N. R. Mendell, and H. C. T. JR., "Probabilistic measure of adequacy of a numerical search for a global maximum," *Journal of the American Statistical Association*, vol. 84, no. 408, pp. 1020–1023, Dec. 1989.

[45] Y. Freund and R. E. Schapire, "A decision-theoretic generalization of on-line learning and an application to Boosting." *Journal of Computer and System Sciences*, vol. 55, no. 1, pp. 119–139, 1997.

[46] J. Friedman, T. Hastie, and R. Tibshirani, "Additive logistic regression: a statistical view of Boosting," *The annals of statistics*, vol. 38, no. 2, pp. 337–374, 2000.

[47] J. Friedmann, E. Fishler, and H. Messer, "General asymptotic analysis of the generalized likelihood ratio test for a gaussian point source under statistical or spatial mismodelling," *IEEE Trans. Signal Process.*, vol. 50, pp. 2617–2631, 11 2002.

[48] A. A. Gaivoronski, "Convergence analysis of parallel backpropagation algorithm for neural networks," *Optim. Methods Softw.*, vol. 4, pp. 117–134, 1994.

[49] L. Gan and J. Jiang, "A test for global maximum," *Journal of the American Statistical Association*, vol. 94, pp. 847–854, Sept. 1999.

[50] P. K. Goel and J. Ginebra, "When is one experiment always better than another," *The statistician*, vol. 52, no. 4, pp. 515–537, 2003.

[51] L. Grippo, "A class of unconstrained minimization methods for neural networks training," *Optim. Methods Softw.*, vol. 4, pp. 135–150, 1994.

[52] ——, "Convergent on-line algorithms for supervised learning in neural networks," *IEEE Trans. Neural Networks*, vol. 11, no. 6, pp. 1284–1299, Nov. 2000.

[53] L. D. Haan, "Estimation of the minimum of a function using order statistics," *J. Amer. Statist. Assoc.*, vol. 76, pp. 467–469, 1981.

[54] T. Hastie, R. Tibshirani, and J. Friedman, *The elements of statistical learning*. New York: Springer-Verlag, 2001.

[55] P. Huber, *Robust Statistics*. New York: John Wiley & Sons, 1981.

[56] D. R. Hunter and K. Lange, "A tutorial on MM algorithms," *The American Statistician*, vol. 58, no. 1, pp. 30–37, Feb 2004.

[57] R. I. Jennrich, "Asymptotic properties of non-linear least squares estimators," *Annals of Mathematical Statistics*, vol. 40, no. 2, pp. 633–643, Apr. 1969.

[58] N. L. Johnson, S. Kotz, and A. Balkrishnan, *Continuous univariate distributions: Vol. 2*. Wiley, New York, 1994.

[59] L. P. Kaelbling, M. Littman, and A. Cassandra, "Planning and acting in partially observable stochastic domains," *Artificial Intelligence*, vol. 101, 1998.

[60] C. K. K. Kastella and A. Hero, "Sensor management using an active sensing approach," *Signal Processing*, vol. 85, no. 3, pp. 607–624, March 2005.

[61] S. Kay, *Fundamentals of Statistical Signal Processing - Estimation Theory*. Prentice Hall, 1993.

[62] M. Kearns, Y. Mansour, and A. Ng, "Approximate planning in large POMDPs via reusable trajectories," in *Advances in Neural Information Processing Systems*, vol. 12. MIT Press, 2000.

[63] R. W. Keener, "Local information and the design of sequential hypothesis tests," *Journal of Statistical Planning and Inference*, vol. 130, no. 1-2, pp. 111–125, 2005.

[64] V. M. Kibardin, "Decomposition into functions in the minimization problem," *Autom. Remote Control*, vol. 40, pp. 1311–1321, 1980.

[65] K. C. Kiwiel, "Convergence of approximate and incremental subgradient methods for convex optimization," *SIAM J. Optim.*, vol. 14, no. 3, pp. 807–840, 2004.

[66] E. Kreyszic, *Advanced engineering mathematics*. New York: John Wiley & Sons, 1988.

[67] H. Krim and M. Viberg, "Two decades of array signal processing," *IEEE signal processing magazine*, pp. 67–94, July 1996.

[68] V. Krishnamurthy, "Algorithms for optimal scheduling and management of hidden markov model sensors," *IEEE Trans. Signal Process.*, vol. 50, no. 6, pp. 1382–1397, June 2002.

[69] S. Kumar, F. Zhao, and D. S. editors, "Special issue on collaborative information processing," *IEEE Signal Processing Magazine*, vol. 19, no. 2, March 2002.

[70] M. Lagoudakis and R. Parr, "Reinforcement learning as classification: Leveraging modern classifiers," in *Proceedings of the Twentieth International Conference on Machine Learning*, 2003.

[71] J. Langford and A. Beygelzimer, "Sensitive error correcting output codes," in *Proceedings of the 18th Annual Conference on Learning Theory*, 2005, pp. 158–172.

[72] J. Langford and B. Zadrozny, "Reducing T-step reinforcement learning to classification," `http://hunch.net/~jl / projects / reductions/reductions.html`, 2003.

[73] ——, "Relating reinforcement learning performance to classification performance," in *Proceedings of the Twenty Second International Conference on Machine Learning*, 2005, pp. 473–480.

[74] B. C. Levy and R. Nikoukhah, "Robust least-squares estimation with a relative entropy constraint," *IEEE Trans. Inform. Theory*, vol. 50, no. 1, pp. 89 – 104, Jan. 2004.

[75] D. Li and Y. H. Hu, "Energy-based collaborative source localization using acoustic microsensor array," *EURASIP Journal on Applied Signal Processing*, no. 4, pp. 321–337, 2003.

[76] Z. Q. Luo, "On the convergence of the LMS algorithm with adaptive learning rate for linear feedforward networks," *Neural Comput.*, vol. 3, pp. 226–245, 1991.

[77] Z. Q. Luo and P. Tseng, "Analysis of an approximate gradient projection method with applications to the backpropagation algorithm," *Optim. Methods Softw.*, vol. 4, pp. 85–101, 1994.

[78] O. L. Mangasarian, "Mathematical programming in neural networks," *ORSA J. Comput.*, vol. 5, pp. 349–360, 1993.

[79] O. L. Mangasarian and M. V. Solodov, "Serial and parallel backpropagation convergence via nonmonotone perturbed minimization," *Optim. Methods Softw.*, vol. 4, pp. 103–116, 1994.

[80] J. Marble, D. Blatt, and A. Hero, "Confirmation sensor scheduling using a reinforcement learning approach," in *SPIE Defense and Security Symposium*, Orlando, Florida, April 2006, to appear.

[81] G. J. McLachlan and T. Krishnan, *The EM Algorithm and Extensions*. John Wiley & Sons, 1997.

[82] R. Meyer and C. Burrus, "A unified analysis of multirate and periodically time-varying digital filters," *IEEE Trans. Cicuits Systems*, vol. 22, no. 3, pp. 162–168, March 1975.

[83] H. Moriyama, N. Yamashita, and M. Fukushima, "The incremental Gauss-Newton algorithm with adaptive stepsize rule," *Comput. Optim. Appl.*, vol. 26, no. 2, pp. 107–141, Nov. 2003.

[84] S. A. Murphy, "A generalization error for Q-learning," *Journal of Machine Learning Research*, vol. 6, pp. 1073–1097, 2005.

[85] V. Nagesha and S. M. Kay, "Maximum likelihood estimation for array processing in colored noise," *IEEE Trans. Signal Process.*, vol. 44, no. 2, pp. 169–180, February 1996.

[86] R. M. Neal and G. E. Hinton, "A view of the EM algorithm that justifies incremental, sparse, and other variants," in *Learning in Graphical Models*, M. I. Jordan, Ed. Dordrecht: Kluwer Academic Publishers, 1994, pp. 355–368.

[87] A. Nedic and D. P. Bertsekas, "Incremental subgradient methods for nondifferentiable optimization," *SIAM J. Optim.*, vol. 12, no. 1, pp. 109–138, 2001.

[88] W. K. Newey, "Maximum likelihood specification testing and conditional moment tests," *Econometrica*, vol. 55, pp. 1047–1070, 1985.

[89] W. K. Newey and K. D. West, "Automatic lag selection in covariance matrix estimation," *Reviews of Econometric Studies*, vol. 61, pp. 631–653, 1994.

[90] R. D. Nowak, "Distributed EM algorithms for density estimation and clustering in sensor networks," *IEEE Trans. Signal Processing*, vol. 51, no. 8, pp. 2245–2253, Aug. 2003.

[91] B. T. Polyak, *Introduction to Optimization*. New York: Optimization Software, Inc., 1987.

[92] J. G. Proakis and D. G. Manolakis, *Digital Signal Processing*. Englewood Cliffs, NJ: Prentice Hall, 1996.

[93] M. L. Puterman, *Markov decision processes: discrete stochastic dynamic programming*. John Wiley & Sons, Inc, 1994.

[94] M. G. Rabbat and R. D. Nowak, "Decentralized source localization and tracking," in *Proceedings of the 2004 IEEE International Conference on Acoustics, Speech, and Signal Processing*, Montreal, Canada, May 2004, pp. 921 – 924.

[95] ——, "Distributed optimization in sensor networks," in *Proceedings of the Third International Symposium on Information Processing in Sensor Networks*. Berkeley, California: ACM Press, New York, April 2004, pp. 20–27.

[96] ——, "Quantized incremental algorithms for distributed optimization," *IEEE Journal on Selected Areas in Communications*, vol. 23, no. 4, pp. 798 – 808, Apr. 2005.

[97] C. R. Rao, *Linear Statistical Inference and Its Applications*. John Wiley & Sons, 1973.

[98] C. Rao and L. Zhao, "Asymptotic behavior of maximum likelihood estimates of superimposed exponential signals," *IEEE Trans. Signal Process.*, vol. 41, no. 3, pp. 1461–1464, March 1993.

[99] W. J. J. Rey, *Introduction to Robust and Quasi-Robust Statistical Methods*. Berlin: Springer-Verlag, 1983.

[100] A. W. Roberts and D. E. Varberg, *Convex Functions*. New York: Academic Press, 1973.

[101] R. T. Rockafeller, *Convex Analysis*. Princeton, NJ: Princeton University Press, 1970.

[102] B. V. Roy, *Handbook of Markov Decision Processes: Methods and Applications*. Kluwer, 2001, ch. Neuro Dynamic Programming: Overview and Recent Trends.

[103] B. Schölkopf and A. J. Smola, *Learning with Kernels*. MIT, Press, 2002.

[104] X. Sheng and Y. H. Hu, "Energy based acoustic source localization," in *Information Processing in Sensor Networks, Second International Workshop*, ser. Lecture Notes in Computer Science, Z. Feng and G. Leonidas, Eds., vol. 2634. Palo Alto, California: Sringer-Verlag, New York, April 2003, pp. 285–300.

[105] ——, "Maximum likelihood multiple-source localization using acoustic energy measurements with wireless sensor networks," *IEEE Trans. Signal Processing*, vol. 53, no. 1, pp. 44–53, Jan. 2005.

[106] S. Slump and B. Hoenders, "The determination of the location of the global maximum of a function in the presence of several local extrema," *IEEE Trans. Inform. Theory*, vol. 31, no. 4, pp. 490 – 497, July 1985.

[107] C. G. Small, J. Wang, and Z. Yang, "Eliminating multiple root problems in estimation," *Statist. Sci.*, vol. 15, no. 4, pp. 313 – 341, 2000.

[108] M. V. Solodov, "Incremental gradient algorithms with stepsizes bounded away from zero," *Comput. Optim. Appl.*, vol. 11, pp. 23–35, 1998.

[109] A. Srinivasan, "The landsat data set," `http://www.niaad.liacc.up.pt/old/ statlog/ datasets/ satimage/ satimage.doc.html`, 1994.

[110] D. Starer and A. Nehorai, "Newton algorithms for conditional and unconditional maximum likelihood estimation of the parameters of exponential signals in noise," *IEEE Trans. Signal Process.*, vol. 40, no. 6, pp. 1528–1534, June 1992.

[111] P. Stoica and A. Nehorai, "Music, maximum likelihood, and Cramér Rao bound," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 37, no. 5, pp. 720–741, May 1989.

[112] R. S. Sutton and A. G. Barto, *Reinforcement Learning*. MIT Press, 1998.

[113] R. Szewczyk, E. Osterweil, J. Polastre, M. Hamilton, A. Mainwaring, and D. Estrin, "Habitat monitoring with sensor networks," *Commun. ACM*, vol. 47, no. 6, pp. 34–40, June 2004.

[114] G. Tauchen, "Diagnostic testing and evaluation of maximum likelihood models," *Journal of Econometrics*, vol. 30, pp. 415–444, 1985.

[115] H. L. V. Trees, *Detection, Estimation, and Modulation Theory*. New York: John Wiley & Sons, 2001.

[116] P. Tseng, "An incremental gradient(-projection) method with momentum term and adaptive stepsize rule," *SIAM J. Optim.*, vol. 8, no. 2, pp. 506–531, May 1998.

[117] P. K. Varshney, *Distributed Detection and Data Fusion*. Springer-Verlag, 1997.

[118] M. R. Veall, "Testing for a glomal maximum in an econometric context," *Econometrica*, vol. 58, no. 6, pp. 1459–1465, Nov. 1990.

[119] M. Vidyasagar, *Learning and Generalization with Applications to Neural Networks*, 2nd ed. Londom: Springer–Verlag, 2003.

[120] H. White, "Nonlinear regression on cross-section data," *Econometrica*, vol. 48, no. 3, pp. 721–746, Apr. 1980.

[121] ——, "Consequences and detection of misspecified nonlinear regression models," *Journal of the American Statistical Association*, vol. 76, no. 374, pp. 419–433, Jun. 1981.

[122] ——, "Maximum likelihood estimation of misspecified models," *Econometrica*, vol. 50, no. 1, pp. 1–26, Jan. 1982.

[123] ——, *Advances in Econometrics*. New York: Cambridge University Press, 1987, ch. Specification Testing in Dynamic Models.

[124] ——, *Estimation, Inference and Specification Analysis*. Cambridge University Press, 1994.

[125] G. Wu, E. K. P. Chong, and R. Givan, "Burst-level congestion control using hindsight optimization," *IEEE Trans. on Automatic Control*, vol. 47, no. 6, pp. 979–991, June 2004.

[126] W. Xu, A. Baggeroer, and K. L. Bell, "A bound on mean-square estimation error with background parameter mismatch," *IEEE Trans. Inform. Theory*, vol. 50, no. 5, pp. 621–632, Apr. 2004.

[127] S. Yau and Y. Bresler, "Maximum likelihood parameter estimation of superimposed signals by dynamic programming," *IEEE Trans. Signal Process.*, vol. 41, no. 2, pp. 804 – 820, Feb. 1993.

[128] I. Ziskind and M. Wax, "Maximum likelihood localization of diversely polarized sources by simulated annealing," *IEEE Trans on Antennas and Propagation*, vol. 38, no. 7, pp. 1111 – 1114, July 1990.