Topics in Genomic Signal Processing

Guido Hugo Jajamovich

Submitted in partial fulfilment of the
requirements for the degree of
Doctor of Philosophy
in the Graduate School of Arts and Sciences

COLUMBIA UNIVERSITY

2012

ABSTRACT

Topics in Genomic Signal Processing

Guido Hugo Jajamovich

Genomic information is digital in its nature and admits mathematical modeling in order to gain biological knowledge. This dissertation focuses on the development and application of detection and estimation theories for solving problems in genomics by describing biological problems in mathematical terms and proposing a solution in this domain. More specifically, a novel framework for hypothesis testing is presented, where it is desired to decide among multiple hypotheses and where each hypothesis involves unknown parameters. Within this framework, a test is developed to perform both detection and estimation jointly in an optimal sense. The proposed test is then applied to the problem of detecting and estimating periodicities in DNA sequences. Moreover, the problem of motif discovery in DNA sequences is presented, where a set of sequences is observed and it is needed to determine which sequences contain instances (if any) of an unknown motif and estimate their positions. A statistical description of the problem is used and a sequential Monte Carlo method is applied for the inference. Finally, the phasing of haplotypes for diploid organisms is introduced, where a novel mathematical model is proposed. The haplotypes that are used to reconstruct the observed genotypes of a group of unrelated individuals are detected and the haplotype pair for each individual in the group is esti-

mated. The model translates a biological principle, the maximum parsimony principle, to a sparseness condition.

Contents

List of Figures

Acknowledgements

First and foremost, I would like to thank my adviser, Professor Xiaodong Wang, who has supported me throughout the doctoral program with his patience and knowledge while allowing me the room to work in my own way. The result of our work for the past four years is contained in this dissertation. I would especially like to express my gratitude to him for paving the way for me to become an independent researcher.

I also want to thank my multiple collaborators throughout the program, as I have learned and experienced different approaches when facing a new problem and proposing a novel solution. I would like to thank specially my friend and collaborator Dr. Ali Tajer with whom I have worked closely in many projects.

I also would like to extend my gratitude to the thesis committee members, Professors Shih-Fu Chang and John Wright, and Dr. Michael Samoilov and Dr. Ta-Hsin Li. I want to thank them for taking time to go over in great detail the works contained in this thesis.

Most importantly, I would like to thank my parents, brother and Lujan. For the past four years, their constant encouragement have kept me focused on the task at hand. Without their support, none of this would have been possible. Finally, I would like to acknowledge my friends for being with me throughout all the highs and lows of research.

Chapter 1

Introduction

Watson and Crick showed that the molecular structure of deoxyribonucleic acid (DNA) has "great simplicity" as it can be described as a double helix consisting of two strands [1]. Each strand is composed of a chain of bases of four types: Adenine (A), Cytosine (C), Guanine (G) and Thymine (T), each with different biochemical properties. The two strands run in opposite directions and are connected to each other by chemical pairing of each base on one strand to a specific partner on the other strand. The pairing is established only between complementary bases also called base pairs; that is, A forms an hydrogen bond with its complementary base T, and C with G. This fact constrains the two strands to consist largely of the same information. Within this description, information in DNA is digital in its nature and is encoded as a sequence where each character belongs to the alphabet $\{A, C, T, G\}$. The analogy with a computer is evident, whereas in the latter case, information is encoded digitally as sequences of zeros and ones.

The central dogma of molecular biology describes how biological infor-

mation is transferred [2, 3]. The normal flow of this information is from DNA to ribonucleic acid (RNA) (transcription) and from RNA to proteins (translation). Proteins are an essential part of organisms and are involved in most of the processes within cells. A protein consists of a sequence of amino acids that is defined by the sequence of bases in a DNA segment called a gene.

The genome comprises of most DNA of an organism containing all the biological information needed to build and maintain that organism. It includes both genes and other segments of DNA. For example, the human genome is comprised of approximately $25,000$ genes and is around 3 billion base pairs long [4]. Each cell of an organism contains a complete copy of the genome, and despite the fact that the genome is the same in each cell, cells actually produce different amounts and types of proteins. Moreover, a cell produces different proteins at different stages within its cycle of life. Protein production is influenced by the internal environment and by signals from other cells. Thus the transcription and translation of genes to proteins is a part of a complex network of interactions involving genes, proteins, and RNA, as well as other factors such as temperature and the presence or absence of nutrients within the cell.

Genomics is an interdisciplinary field concerned with the study of the genomes of organisms. This discipline has four main goals [5]. First, it aims at determining the sequence of bases of the genomes of different species and finding differences and similarities among the different species. Traditionally, the sequences have been read using Sanger's method [6] with an accuracy of 99.999%, but with limited level of parallelization [7]. The need

for faster and cheaper methods has led to the advent of next-generation sequencing technologies that parallelized the reading process at the expense of accuracy [7]. These technologies read short subsequences that need to be assembled to determine the whole DNA sequence by solving ambiguous repeat regions. Sequencers from 454 Life Sciences/Roche, Solexa/Illumina and Applied Biosystems (SOLiD technology) read subsequences of $35 - 40$ base pairs long [8]. This progress in high-throughput platforms is moving towards an era of synthetic genomics and personalized medicine [9].

Second, the genes need to be discovered and the function of the associated proteins uncovered. Given the vast wealth of DNA sequences produced by the sequencing platforms, it is required to identify the segments that correspond to genes. Methods to find genes can be classified as those that use a single genome and those that utilize a comparative approach, where information about one organism is used to understand another related one. In particular, methods that rely on only one genome make use of particular properties of genes, like its statistical tendencies concerning the distribution of triplet of DNA bases [5].

Third, it is important to understand how genes and proteins interact in order to control cellular processes. It is known that different cells have a copy of the same genome, but each cell produces a different set of proteins. Moreover, the set changes over time, even though the genome continues to be the same. The transcription and translation of genes is typically controlled by complex networks of regulatory interactions which involves proteins attaching to highly specific nucleic acid sequences activating or repressing the amount of protein generated by a given gene. These networks

need to be uncovered in order to understand the underlying mechanisms used by cells.

Fourth, genomics aims at discovering associations between gene mutations and diseases. The genomes of any two organisms, even within a same species, differ considerably as DNA sequences present variations. There are multiple sources for these variations, such as mutations, leading to susceptibility to diseases. For example, sickle-cell disease is an inherited hemoglobin disorder characterized by red blood cells that assume an abnormal, less malleable, sickle shape and occurs because of a mutation in the hemoglobin gene [10].

This thesis focuses on solving problems in genomics by proposing novel statistical and mathematical models, where the problem in hand can be stated as a detection and estimation problem. In most of the aforementioned problems, we have observations and we need, based on these observations, to detect and estimate among different scenarios with unknown parameters.

1.1 Thesis Overview

In Chapter 2, the problem of detecting and estimating periodicities in DNA sequences is introduced. DNA sequences present numerous types of regularities and repetitions related with the underlying structure of the sequences, e.g., a periodicity of 21 bases is linked with α-helix formation protein molecules [11] and a periodicity of three is identified with protein coding regions of the DNA. In order to perform an optimal detection and estimation of these periodicities, a novel framework is introduced for composite binary

hypothesis testing, where the objective is to decide between two hypotheses each of which involves unknown parameters of interest and to be estimated. The existing approaches on composite hypothesis testing place the primary emphasis on the detection part by solving this part optimally and treating the estimation part suboptimally. The proposed framework, in contrast, treats both problems simultaneously and in a jointly optimal manner. The resulting test exhibits the flexibility to achieve any desired balance between the detection and estimation performances. By exploiting this flexibility, depending on the application in hand, this new technique offers the freedom to put different emphasis on the detection and estimation subproblems. The proposed optimal joint detection and estimation framework is also extended to multiple composite hypothesis test. The proposed test is then applied to the problem of detecting and estimating periodicities in DNA sequences, where it is shown the advantages of the new framework compared to the classical Neyman-Pearson approach and the GLRT [12].

In Chapter 3, the problem of motif discovery in DNA sequences is examined. Motifs occur in many places within the genome, and they usually carry evolutionary or functional significance. In particular, genes with instances of a motif nearby often indicate that they are being regulated by the same protein, which can reveal gene regulatory relationships. In a motif discovery problem, we are given a set of DNA sequences to discover a common motif that is shared within these sequences. A priori knowledge of such motif features as length or composition is likely to either be incomplete, uncertain, or even entirely absent. We consider the number, length, and locations of individual motif instances in each sequence to be unobserv-

able directly. The available data consists solely of the sequences themselves, wherein motif patterns of interest, which remain to be discovered, may (or may not) be embedded. The proposed approach for motif discovery is based on a statistical description of the problem within a Bayesian framework and it outperforms the traditional motif discovery approaches [13].

Finally, the problem of haplotype inference is presented in Chapter 4. A haplotype is the set of bases in DNA sequences where variations are known to happen. In diploid organisms, haplotypes come in pairs, one inherited from each parent, and the knowledge of the haplotype pair of an individual can be used to predict diseases, help designing drugs and it is key to the development of personalized medicine. However, experimentally determining haplotypes is expensive and time-consuming, so genotypes are usually measured instead. A genotype is a single set representation of the haplotype pair that needs to be phased in order to recover the haplotype pairs, which is not possible if only one genotype is observed. However, given the set of genotypes for a group of unrelated individuals, it is possible to infer the haplotype pair for each individual based on side-information from population genetics: the maximum parsimony principle. Two related formulations of the haplotype inference problem are proposed that translate the maximum parsimony principle into the sparse representation of genotypes. The proposed solutions are tested with different data sets and the performances are compared with the state-of-the-art methods, achieving similar or better results [14].

Each chapter is self-contained, has its own notation and can be read independently.

Chapter 2

Optimal Detection and Estimation: Discovering Periodicities in DNA Sequences

2.1 Introduction

Applications that involve simultaneous detection and parameter estimation are frequently found in practice. Composite hypothesis testing involves making a decision among multiple hypotheses, and upon deciding in favor of one hypothesis, also making an estimate of some *unknown* parameters associated with that hypothesis. It has applications in a broad range of areas such as wireless communications, genetics, neuroscience and finance.

It is well established how to solve, optimally, the detection problem and the estimation problem separately. But little is known as to how to treat the joint detection and estimation problem optimally. These two subproblems in composite hypothesis testing can be solved separately in a decoupled

manner, where given a constraint on the probability of false alarm, detection is performed first to decide among the different hypotheses by using the Neyman-Pearson test [15] that achieves the optimal detection performance. Then, Bayesian estimation can be employed to estimate the parameters associated with the hypothesis resulted from the detection step without taking into account the uncertainty of the detection step. Such an approach guarantees the optimal detection performance but there is no control over the estimation performance.

Another common approach to composite hypothesis testing is the well-known generalized likelihood ratio test (GLRT). This test first performs the maximum-likelihood (ML) estimation of all unknown parameters and then replaces the unknown parameters with their ML estimates, transforming the original problem into a simple hypothesis testing problem. This latter detection problem is then solved through the likelihood ratio test such that a constraint on false alarm probability is satisfied. In GLRT, the primary emphasis is on the detection performance and the estimation performance is treated as a secondary performance measure. This test offers no flexibility in terms of detection and estimation performance tradeoff. Moreover, it is known that the GLRT is not always optimal [16, 17] in a Neyman-Pearson sense; that is, among the decision rules with a constraint on the probability of false alarm, it does not necessarily minimize the probability of miss detection. However, optimality results are known for this test in the limiting case of a large number of observations [18]. Note that when the hypotheses are simple ones, the GLRT becomes the Neyman-Pearson test.

An alternative approach is given in [19], where a test is developed that is

optimal under the Bayesian criterion, i.e., in the sense of minimum average risk under different coupling schemes between the detection and estimation tasks. The test assumes, however, that the two hypotheses correspond to a signal in noise and noise alone respectively, and allows only the signal to contain unknown parameters. In [20], the theory in [19] is extended to the multiple hypothesis testing case, where the unknown parameters need to be energy-type parameters, e.g, amplitude and duration of a signal. Yet another approach was introduced in [21], where the error probabilities under the two hypotheses are replaced by estimation costs. The test is found by constraining the estimation cost under the nominal hypothesis while optimizing the corresponding cost under the alternative hypothesis.

In [22], a multi-hypothesis test is proposed based on the worst-case estimation and worst-case detection performances subject to a false-alarm constraint. The unknown parameters are fixed, nonrandom and belong to a finite discrete set, which makes it possible to convert the estimation subproblem to an extra detection subproblem. More recently, in [23], the combined problem is treated for the case that only the nominal hypothesis has unknown parameters and all parameters in the alternative hypothesis are known. A new test is proposed based on an optimization formulation with an objective function that is associated with the estimation performance and with constraints on the detection performance. In deriving the test, the fact that unknown parameters are associated with only one hypothesis plays a crucial role, which makes it possible to show that the constraints on the detection performance are achieved with equality. This fact further simplifies the objective estimation performance measure, making the extension to

the general case with unknown parameters in both hypotheses a nontrivial problem.

In this chapter we allow both hypotheses to have unknown parameters, in order to develop the general theory of optimal joint detection and estimation. We further extend the proposed framework to the general multiple hypothesis testing problem with unknown parameters associated with each hypothesis. The proposed optimal test provides the freedom to trade off detection and estimation accuracies. As an application of the proposed theory, we solve the problem of detecting and estimating periodicities in DNA sequences.

The remainder of the chapter is organized as follows. Section 2.2 introduces the composite hypothesis test and formulates the optimal joint detection and estimation problem. We develop the general theory in Section 2.3 and apply it to the periodicity detection and estimation in DNA sequences in Section 2.4.

2.2 Joint Detection and Estimation

2.2.1 Composite Hypothesis Test

Let $\boldsymbol{X}$ be an observation signal and consider the following *composite* binary hypothesis testing problem:

$$\begin{aligned} & \mathsf{H}_0: \quad \boldsymbol{X} \sim f_0(\boldsymbol{X} \mid \boldsymbol{\theta}_0), \quad \text{with} \quad \boldsymbol{\theta}_0 \sim \pi_0(\boldsymbol{\theta}_0), \\ \text{and} \quad & \mathsf{H}_1: \quad \boldsymbol{X} \sim f_1(\boldsymbol{X} \mid \boldsymbol{\theta}_1), \quad \text{with} \quad \boldsymbol{\theta}_1 \sim \pi_1(\boldsymbol{\theta}_1), \end{aligned} \tag{2.1}$$

where $f_i(\boldsymbol{X} \mid \boldsymbol{\theta}_i)$ and $\pi_i(\boldsymbol{\theta}_i)$ are known probability density functions (pdfs) for $i \in \{0, 1\}$. Under hypothesis H_i the distribution of the observation belongs to an ensemble of distributions $f_i(\boldsymbol{X} \mid \boldsymbol{\theta}_i)$ specified by random parameter $\boldsymbol{\theta}_i$ with the prior distribution $\pi_i(\boldsymbol{\theta}_i)$. We wish to develop a mechanism that decides between H_0 and H_1 reliably and furthermore, when it decides in favor of H_i also provides an accurate estimate of the related unknown parameter $\boldsymbol{\theta}_i$.

Both the GLRT and a Neyman-Person test followed by a Bayesian estimation solve the above combined detection and estimation problem by decomposing the joint problem into two subproblems and solving each optimally. For instance, in the latter case, the Neyman-Pearson optimum test is used for detection and the optimum Bayesian estimator is used for parameter estimation. Treating each subproblem independently does not necessarily yield the optimum overall performance. Both approaches are not capable of emphasizing either subproblem according to the need of the specific application.

Here we formulate the combined problem in a more natural way by posing the combined detection and estimation tasks in a way that captures both detection and estimation accuracies. In particular, we aim to minimize an estimation-pertinent cost subject to appropriate constraints on the tolerable levels of detection errors, i.e., miss detection and false alarm error probabilities. The main feature of this approach is that it provides the freedom to strike any desired balance between the detection and estimation performances.

2.2.2 Background

One approach for designing a test when we are interested in *only* the detection performance but *not* the estimation performance is the Neyman-Pearson method [15], which maximizes the detection probability given a constraint on the probability of false alarm. Therefore, in this approach, the estimation performance is suboptimal in favor of achieving the *optimal* detection performance. The optimal Neyman-Pearson test is given in the following lemma.

Lemma 1 (Neyman-Pearson) *The test that maximizes the detection probability subject to an upper bound on the false alarm probability is*

$$\frac{f_1(\boldsymbol{X})}{f_0(\boldsymbol{X})} \underset{\mathsf{H}_0}{\overset{\mathsf{H}_1}{\gtrless}} \lambda_{\mathrm{NP}} \ ,$$

where

$$f_i(\boldsymbol{X}) = \int f_i(\boldsymbol{X} \mid \boldsymbol{\theta}_i)\ \pi_i(\boldsymbol{\theta}_i)\ d\boldsymbol{\theta}_i \ , \quad \textit{for} \quad i \in \{0,1\} \ ,$$

and the threshold λ_{NP} is selected to satisfy the false alarm constraint with equality.

This test does not consider the estimation performance. More specifically, it first carries out the detection test and if it decides in favor of H_i, in the second step it provides an estimate for $\boldsymbol{\theta}_i$, given that it has decided the true hypothesis is H_i. This two-step approach is *not* optimal from the joint detection and estimation point of view. In this work, we look for an alternative test that takes into account both detection and estimation qualities and is *optimal* in that sense.

2.2.3 Definitions

In order to decide between the two hypotheses, we adopt the class of randomized tests. Given the observation $\boldsymbol{X}$, we assign the probabilities $\delta_0(\boldsymbol{X})$ and $\delta_1(\boldsymbol{X})$ to accept hypotheses H_0 and H_1, respectively. As $\delta_0(\boldsymbol{X})$ and $\delta_1(\boldsymbol{X})$ are probabilities, we require that $\delta_0(\boldsymbol{X}), \delta_1(\boldsymbol{X}) \geq 0$ and moreover, as we always decide between the two hypotheses, the randomized test needs to satisfy $\delta_0(\boldsymbol{X}) + \delta_1(\boldsymbol{X}) = 1$. Note that classical deterministic tests are special cases of randomized tests. Furthermore, we denote the true hypothesis and the decision of the detector by $\mathsf{T} \in \{\mathsf{H}_0, \mathsf{H}_1\}$ and $\mathsf{D} \in \{\mathsf{H}_0, \mathsf{H}_1\}$, respectively. Therefore, given the randomized tests $\{\delta_0(\boldsymbol{X}), \delta_1(\boldsymbol{X})\}$, the Type-I and Type-II detection error probabilities are

$$\begin{aligned} \mathsf{P}_1(\delta_0, \delta_1) &\triangleq \mathsf{P}(\mathsf{D} = \mathsf{H}_1 \mid \mathsf{T} = \mathsf{H}_0) \\ \text{and} \quad \mathsf{P}_2(\delta_0, \delta_1) &\triangleq \mathsf{P}(\mathsf{D} = \mathsf{H}_0 \mid \mathsf{T} = \mathsf{H}_1) \ , \end{aligned} \tag{2.2}$$

respectively. Once we decide that the observation $\boldsymbol{X}$ is drawn from hypothesis H_i, we are also interested in providing an estimate $\hat{\boldsymbol{\theta}}_i(\boldsymbol{X})$ for $\boldsymbol{\theta}_i$. In order to capture the estimation quality, we assign the non-negative costs $\mathsf{C}_0(\hat{\boldsymbol{\theta}}_0(\boldsymbol{X}), \boldsymbol{\theta}_0)$ and $\mathsf{C}_1(\hat{\boldsymbol{\theta}}_1(\boldsymbol{X}), \boldsymbol{\theta}_1)$ to the estimators $\hat{\boldsymbol{\theta}}_0(\boldsymbol{X})$ for $\boldsymbol{\theta}_0$ and $\hat{\boldsymbol{\theta}}_1(\boldsymbol{X})$ for $\boldsymbol{\theta}_1$ respectively. Two popular cost functions corresponding to the minimum mean-squared error (MMSE) and maximum a-posteriori (MAP) esti-

mation criteria are

$$\text{MMSE}: \quad \mathsf{C}(\hat{\boldsymbol{\theta}}(\boldsymbol{X}), \boldsymbol{\theta}) = \|\hat{\boldsymbol{\theta}}(\boldsymbol{X}) - \boldsymbol{\theta}\|^2 \ ,$$

$$\text{and} \quad \text{MAP}: \quad \mathsf{C}(\hat{\boldsymbol{\theta}}(\boldsymbol{X}), \boldsymbol{\theta}) = \begin{cases} 0 & \text{if } \ \|\hat{\boldsymbol{\theta}}(\boldsymbol{X}) - \boldsymbol{\theta}\| \leq \delta \ll 1, \\ 1 & \text{otherwise.} \end{cases}$$

For given cost functions $\mathsf{C}_0(\hat{\boldsymbol{\theta}}_0(\boldsymbol{X}), \boldsymbol{\theta}_0)$ and $\mathsf{C}_1(\hat{\boldsymbol{\theta}}_1(\boldsymbol{X}), \boldsymbol{\theta}_1)$, define the following average *posterior* cost functions, given by

$$\mathsf{C}_{i,\mathrm{p}}(\hat{\boldsymbol{\theta}}_i(\boldsymbol{X}) \mid \boldsymbol{X}) \stackrel{\triangle}{=} \mathsf{E}_{\boldsymbol{\theta}_i}[\mathsf{C}_i(\hat{\boldsymbol{\theta}}_i(\boldsymbol{X}), \boldsymbol{\theta}_i) \mid \boldsymbol{X}], \quad i = 0, 1 \ , \tag{2.3}$$

where the expectation is with respect to $\boldsymbol{\theta}_i$. Therefore, the minimum average posterior cost is

$$\mathsf{C}^*_{i,\mathrm{p}}(\boldsymbol{X}) \stackrel{\triangle}{=} \inf_{\boldsymbol{U}} \mathsf{C}_{i,\mathrm{p}}(\boldsymbol{U} \mid \boldsymbol{X}), \tag{2.4}$$

and the minimizer of the posterior cost, which is the well-known Bayesian estimator, is [15, pp. 142]

$$\hat{\boldsymbol{\theta}}^*_i(\boldsymbol{X}) \stackrel{\triangle}{=} \arg \inf_{\hat{\boldsymbol{\theta}}_i(\boldsymbol{X})} \mathsf{C}_{i,\mathrm{p}}(\hat{\boldsymbol{\theta}}_i(\boldsymbol{X}) \mid \boldsymbol{X}) \ . \tag{2.5}$$

2.2.4 Problem Formulation

Given two non-negative cost functions $\mathsf{C}_0(\hat{\boldsymbol{\theta}}_0(\boldsymbol{X}), \boldsymbol{\theta}_0)$ and $\mathsf{C}_1(\hat{\boldsymbol{\theta}}_1(\boldsymbol{X}), \boldsymbol{\theta}_1)$, we are interested in providing an estimate for $\boldsymbol{\theta}_i$ only when we decide in favor of H_i. Therefore, the estimation cost in estimating $\boldsymbol{\theta}_i$ is meaningful only when we accept hypothesis H_i. Hence, to characterize the performance measure,

we consider only the average estimation cost for estimating $\boldsymbol{\theta}_i$ under hypothesis H_i when deciding in favor of H_i, which for a given randomized policy $\{\delta_0(\boldsymbol{X}), \delta_1(\boldsymbol{X})\}$ and estimator $\hat{\boldsymbol{\theta}}_i(\boldsymbol{X})$ is given by

$$\mathcal{L}_i(\delta_0, \delta_1, \hat{\boldsymbol{\theta}}_i) \triangleq \mathsf{E}_i[\mathsf{C}_i(\hat{\boldsymbol{\theta}}_i(\boldsymbol{X}), \boldsymbol{\theta}_i) \mid \mathsf{D} = \mathsf{H}_i] \ , \tag{2.6}$$

where the expectation is with respect to $\boldsymbol{X}$ and $\boldsymbol{\theta}_i$. In order to capture the estimation quality of both parameters $\boldsymbol{\theta}_0$ and $\boldsymbol{\theta}_1$, we propose to characterize the overall performance measure using the maximum of these two average estimation costs. Hence, for a given randomized policy $\{\delta_0(\boldsymbol{X}), \delta_1(\boldsymbol{X})\}$ and estimators $\{\hat{\boldsymbol{\theta}}_0(\boldsymbol{X}), \hat{\boldsymbol{\theta}}_1(\boldsymbol{X})\}$ the performance measure to be optimized is[1]

$$\mathcal{L}(\delta_0, \delta_1, \hat{\boldsymbol{\theta}}_0, \hat{\boldsymbol{\theta}}_1) \triangleq \max_{i \in \{0,1\}} \mathcal{L}_i(\delta_0, \delta_1, \hat{\boldsymbol{\theta}}_i) \ . \tag{2.7}$$

The above performance measure only accounts for the estimation performance. In order to incorporate the detection performance we impose upper bound constraints on the detection error probabilities as

$$\begin{aligned} \mathsf{P}_1(\delta_0, \delta_1) &\triangleq \int_{\boldsymbol{X}} \delta_1(\boldsymbol{X}) f_0(\boldsymbol{X}) d\boldsymbol{X} \leq \alpha, \\ \text{and} \quad \mathsf{P}_2(\delta_0, \delta_1) &\triangleq \int_{\boldsymbol{X}} \delta_0(\boldsymbol{X}) f_1(\boldsymbol{X}) d\boldsymbol{X} \leq \beta, \ \text{ for } \ \alpha, \beta \in (0,1). \end{aligned} \tag{2.8}$$

Hence, the joint problem of determining the optimal detection rules $\{\delta_0(\boldsymbol{X}), \delta_1(\boldsymbol{X})\}$ and estimators $\{\hat{\boldsymbol{\theta}}_0(\boldsymbol{X}), \hat{\boldsymbol{\theta}}_1(\boldsymbol{X})\}$ is

[1] In the remainder of the chapter we often replace $\delta_i(\boldsymbol{X})$ and $\hat{\boldsymbol{\theta}}_i(\boldsymbol{X})$ by δ_i and $\hat{\boldsymbol{\theta}}_i$, respectively, for notational simplicity.

$$\mathcal{P}(\alpha,\beta) \triangleq \begin{cases} \min_{\{\delta_0,\delta_1,\hat{\boldsymbol{\theta}}_0,\hat{\boldsymbol{\theta}}_1\}} & \mathcal{L}(\delta_0,\delta_1,\hat{\boldsymbol{\theta}}_0,\hat{\boldsymbol{\theta}}_1), \\ \text{s.t.} & \mathsf{P}_1(\delta_0,\delta_1) \leq \alpha, \\ & \mathsf{P}_2(\delta_0,\delta_1) \leq \beta, \end{cases} \tag{2.9}$$

where we also have the implicit constraints $\delta_0(\boldsymbol{X})+\delta_1(\boldsymbol{X}) = 1$ and $\delta_0(\boldsymbol{X}), \delta_1(\boldsymbol{X}) \geq 0$.

2.3 Optimum Joint Detection and Estimation

In this section we obtain the optimal choices of the detection rules $\{\delta_0(\boldsymbol{X}), \delta_1(\boldsymbol{X})\}$ and the estimators $\{\hat{\boldsymbol{\theta}}_0(\boldsymbol{X}), \hat{\boldsymbol{\theta}}_1(\boldsymbol{X})\}$ that solve the problem $\mathcal{P}(\alpha,\beta)$ given in (2.9). In order to proceed we provide the following remarks. Note that there exists an inherent tradeoff between the estimation and detection performances as the detection (estimation) performance can be traded in favor of achieving a better estimation (detection) performance. First we note that the constraints $\mathsf{P}_1(\delta_0,\delta_1) \leq \alpha$ and $\mathsf{P}_2(\delta_0,\delta_1) \leq \beta$ are not necessarily always feasible simultaneously. The following remark provides conditions for the feasibility of the pair (α,β).

Remark 1 (Feasibility) *For the given constraint* $\mathsf{P}_1(\delta_0,\delta_1) \leq \alpha$*, the Type-II detection error* $\mathsf{P}_2(\delta_0,\delta_1)$ *is known to be minimized by the Neyman-Pearson test. Let us define* $\beta^*(\alpha)$ *as the corresponding minimum of* $\mathsf{P}_2(\delta_0,\delta_1)$*. Hence, the two constraints* $\mathsf{P}_1(\delta_0,\delta_1) \leq \alpha$ *and* $\mathsf{P}_2(\delta_0,\delta_1) \leq \beta$ *are feasible simultaneously if and only if*

$$\beta \geq \beta^*(\alpha). \tag{2.10}$$

Remark 2 *The proposed framework of joint detection and estimation trades*

the detection quality, by tolerating a detection error probability that is higher than that is achievable by the Neyman-Pearson test, in favor of enhancing the estimation quality. Allowing for such tradeoff between estimation and detection qualities offers the freedom of putting appropriate emphasis on either the detection or the estimation part, depending on the application.

We find the solution to $\mathcal{P}(\alpha, \beta)$ by finding the optimal estimators $\{\hat{\boldsymbol{\theta}}_0(\boldsymbol{X}), \hat{\boldsymbol{\theta}}_1(\boldsymbol{X})\}$ for fixed detection rules $\{\delta_0(\boldsymbol{X}), \delta_1(\boldsymbol{X})\}$ and then obtaining the optimal choices of the detection rules. In other words, we find the solution of $\mathcal{P}(\alpha, \beta)$ by solving

$$\mathcal{P}(\alpha,\beta) \stackrel{\triangle}{=} \begin{cases} \min_{\{\delta_0,\delta_1\}} & \tilde{\mathcal{L}}(\delta_0,\delta_1), \\ \text{s.t.} & \mathsf{P}_1(\delta_0,\delta_1) \leq \alpha, \\ & \mathsf{P}_2(\delta_0,\delta_1) \leq \beta, \end{cases} \tag{2.11}$$

where

$$\tilde{\mathcal{L}}(\delta_0,\delta_1) \stackrel{\triangle}{=} \min_{\{\hat{\boldsymbol{\theta}}_0,\hat{\boldsymbol{\theta}}_1\}} \mathcal{L}(\delta_0,\delta_1,\hat{\boldsymbol{\theta}}_0,\hat{\boldsymbol{\theta}}_1) \ . \tag{2.12}$$

2.3.1 Estimation

The optimal estimators for fixed detection rules $\{\delta_0(\boldsymbol{X}), \delta_1(\boldsymbol{X})\}$ are found as the minimizers of the function $\mathcal{L}(\delta_0, \delta_1, \hat{\boldsymbol{\theta}}_0, \hat{\boldsymbol{\theta}}_1)$, which are characterized by the following theorem.

Theorem 1 *The solution to the optimization problem*

$$(\hat{\boldsymbol{\theta}}_0^*, \hat{\boldsymbol{\theta}}_1^*) = \arg \min_{\{\hat{\boldsymbol{\theta}}_0,\hat{\boldsymbol{\theta}}_1\}} \mathcal{L}(\delta_0,\delta_1,\hat{\boldsymbol{\theta}}_0,\hat{\boldsymbol{\theta}}_1)$$

is

$$\hat{\boldsymbol{\theta}}_i^*(\boldsymbol{X}) \triangleq \arg \inf_{\hat{\boldsymbol{\theta}}_i(\boldsymbol{X})} \mathsf{C}_{i,\mathrm{p}}(\hat{\boldsymbol{\theta}}_i(\boldsymbol{X}) \mid \boldsymbol{X}) \ , \tag{2.13}$$

and

$$\tilde{\mathcal{L}}(\delta_0, \delta_1) = \min_{\hat{\boldsymbol{\theta}}_0, \hat{\boldsymbol{\theta}}_1} \mathcal{L}(\delta_0, \delta_1, \hat{\boldsymbol{\theta}}_0, \hat{\boldsymbol{\theta}}_1) \quad = \quad \max_{i \in \{0,1\}} \left\{ \frac{\int_{\boldsymbol{X}} \delta_i(\boldsymbol{X}) \ f_i(\boldsymbol{X}) \ \mathsf{C}_{i,\mathrm{p}}^*(\boldsymbol{X}) \ d\boldsymbol{X}}{\int_{\boldsymbol{X}} \delta_i(\boldsymbol{X}) f_i(\boldsymbol{X}) \ d\boldsymbol{X}} \right\}.$$

Proof: From (2.6) and (2.7) recall that

$$\mathcal{L}(\delta_0, \delta_1, \hat{\boldsymbol{\theta}}_0, \hat{\boldsymbol{\theta}}_1) = \max_{i \in \{0,1\}} \mathcal{L}_i(\delta_0, \delta_1, \hat{\boldsymbol{\theta}}_i) \ . \tag{2.14}$$

Let $\mathcal{M}(\Omega, \delta_0, \delta_1, \hat{\boldsymbol{\theta}}_0, \hat{\boldsymbol{\theta}}_1)$ be the convex combination of $\mathcal{L}_0(\delta_0, \delta_1, \hat{\boldsymbol{\theta}}_0)$ and $\mathcal{L}_1(\delta_0, \delta_1, \hat{\boldsymbol{\theta}}_1)$, $\Omega \in [0, 1]$, that is,

$$\mathcal{M}(\Omega, \delta_0, \delta_1, \hat{\boldsymbol{\theta}}_0, \hat{\boldsymbol{\theta}}_1) \triangleq \left\{ \Omega \mathcal{L}_0(\delta_0, \delta_1, \hat{\boldsymbol{\theta}}_0) + (1 - \Omega) \, \mathcal{L}_1(\delta_0, \delta_1, \hat{\boldsymbol{\theta}}_1) \right\}. \tag{2.15}$$

Then (2.14) can be rewritten as a function of $\mathcal{L}(\Omega, \delta_0, \delta_1, \hat{\boldsymbol{\theta}}_0, \hat{\boldsymbol{\theta}}_1)$ as

$$\begin{aligned} \mathcal{L}(\delta_0, \delta_1, \hat{\boldsymbol{\theta}}_0, \hat{\boldsymbol{\theta}}_1) &= \max_{0 \le \Omega \le 1} \mathcal{M}(\Omega, \delta_0, \delta_1, \hat{\boldsymbol{\theta}}_0, \hat{\boldsymbol{\theta}}_1) \\ &= \mathcal{M}(\Omega^*, \delta_0, \delta_1, \hat{\boldsymbol{\theta}}_0, \hat{\boldsymbol{\theta}}_1), \end{aligned} \tag{2.16}$$

where $\Omega^* = 1$ if $\mathcal{L}_0(\delta_0, \delta_1, \hat{\boldsymbol{\theta}}_0) \ge \mathcal{L}_1(\delta_0, \delta_1, \hat{\boldsymbol{\theta}}_1)$, and $\Omega^* = 0$ otherwise.

In what follows, we will first show that for a given Ω, we have

$$\begin{aligned} \min_{\hat{\boldsymbol{\theta}}_0, \hat{\boldsymbol{\theta}}_1} \mathcal{M}(\Omega, \delta_0, \delta_1, \hat{\boldsymbol{\theta}}_0, \hat{\boldsymbol{\theta}}_1) &= \mathcal{M}(\Omega, \delta_0, \delta_1, \hat{\boldsymbol{\theta}}_0^*, \hat{\boldsymbol{\theta}}_1^*) \\ &= \Omega \mathcal{L}_0(\delta_0, \delta_1, \hat{\boldsymbol{\theta}}_0^*) + (1 - \Omega) \, \mathcal{L}_1(\delta_0, \delta_1, \hat{\boldsymbol{\theta}}_1^*), \end{aligned} \tag{2.17}$$

where $\hat{\boldsymbol{\theta}}_0^*$ and $\hat{\boldsymbol{\theta}}_1^*$ are defined in (2.13). We then show that

$$\min_{\hat{\boldsymbol{\theta}}_0,\hat{\boldsymbol{\theta}}_1} \max_{0\leq\Omega\leq 1} \mathcal{M}(\Omega,\delta_0,\delta_1,\hat{\boldsymbol{\theta}}_0,\hat{\boldsymbol{\theta}}_1) = \max_{0\leq\Omega\leq 1} \min_{\hat{\boldsymbol{\theta}}_0,\hat{\boldsymbol{\theta}}_1} \mathcal{M}(\Omega,\delta_0,\delta_1,\hat{\boldsymbol{\theta}}_0,\hat{\boldsymbol{\theta}}_1); \quad (2.18)$$

and from these two results we conclude the proof.

In order to show (2.17), note that

$$\begin{aligned} \mathcal{L}_i(\delta_0,\delta_1,\hat{\boldsymbol{\theta}}_i) &= \mathsf{E}_i[\mathsf{C}_i(\hat{\boldsymbol{\theta}}_i(\boldsymbol{X}),\boldsymbol{\theta}_i) \mid \mathsf{D}=\mathsf{H}_i] \\ &= \frac{\mathsf{E}_i[\delta_i(\boldsymbol{X})\mathsf{C}_i(\hat{\boldsymbol{\theta}}_i(\boldsymbol{X}),\boldsymbol{\theta}_i)]}{\mathsf{P}_i\left(\mathsf{D}=\mathsf{H}_i\right)}, \end{aligned} \quad (2.19)$$

where

$$\mathsf{P}_i(\mathsf{D}=\mathsf{H}_i) = \int_{\boldsymbol{X}} \delta_i(\boldsymbol{X})\ f_i(\boldsymbol{X})\ d\boldsymbol{X}\ . \quad (2.20)$$

We have the following lower bound on $\mathsf{E}_i[\delta_i(\boldsymbol{X})\mathsf{C}_i(\hat{\boldsymbol{\theta}}_i(\boldsymbol{X}),\boldsymbol{\theta}_i)]$ for any given decision rule $\hat{\theta}_i(\boldsymbol{X})$, for $i\in\{0,1\}$.

$$\begin{aligned} \mathsf{E}_i[\delta_i(\boldsymbol{X})\mathsf{C}_i(\hat{\boldsymbol{\theta}}_i(\boldsymbol{X}),\boldsymbol{\theta}_i)] &= \int_{\boldsymbol{\theta}_i}\int_{\boldsymbol{X}} \delta_i(\boldsymbol{X})\mathsf{C}_i(\hat{\boldsymbol{\theta}}_i(\boldsymbol{X}),\boldsymbol{\theta}_i) f_i(\boldsymbol{X}\mid\boldsymbol{\theta}_i)\pi_i(\boldsymbol{\theta}_i)d\boldsymbol{X}d\boldsymbol{\theta}_i \\ &= \int_{\boldsymbol{X}} \delta_i(\boldsymbol{X}) \int_{\boldsymbol{\theta}_i}\Big(\mathsf{C}_i(\hat{\boldsymbol{\theta}}_i(\boldsymbol{X}),\boldsymbol{\theta}_i) f_i(\boldsymbol{X}\mid\boldsymbol{\theta}_i)\pi_i(\boldsymbol{\theta}_i)d\boldsymbol{\theta}_i\Big)d\boldsymbol{X} \\ &= \int_{\boldsymbol{X}} \delta_i(\boldsymbol{X}) f_i(\boldsymbol{X}) \int_{\boldsymbol{\theta}_i}\Big(\mathsf{C}_i(\hat{\boldsymbol{\theta}}_i(\boldsymbol{X}),\boldsymbol{\theta}_i)\pi_i(\boldsymbol{\theta}_i\mid\boldsymbol{X})d\boldsymbol{\theta}_i\Big)d\boldsymbol{X} \\ &= \int_{\boldsymbol{X}} \delta_i(\boldsymbol{X}) f_i(\boldsymbol{X}) \mathsf{E}_{\boldsymbol{\theta}_i}[\mathsf{C}(\hat{\boldsymbol{\theta}}_i(\boldsymbol{X}),\boldsymbol{\theta}_i)\mid\boldsymbol{X}]d\boldsymbol{X} \\ &= \int_{\boldsymbol{X}} \delta_i(\boldsymbol{X}) f_i(\boldsymbol{X}) \mathsf{C}_{i,\mathrm{p}}(\hat{\boldsymbol{\theta}}_i(\boldsymbol{X})\mid\boldsymbol{X})d\boldsymbol{X} \\ &\geq \int_{\boldsymbol{X}} \delta_i(\boldsymbol{X}) f_i(\boldsymbol{X}) \inf_{\boldsymbol{U}} \mathsf{C}_{i,\mathrm{p}}(\boldsymbol{U}\mid\boldsymbol{X})d\boldsymbol{X} \\ &= \int_{\boldsymbol{X}} \delta_i(\boldsymbol{X}) f_i(\boldsymbol{X}) \mathsf{C}^*_{i,\mathrm{p}}(\boldsymbol{X})d\boldsymbol{X}\ . \end{aligned} \quad (2.21)$$

For each term of (2.17), these lower bounds can be achieved by setting the estimators as

$$\hat{\boldsymbol{\theta}}_i^*(\boldsymbol{X}) \triangleq \arg \inf_{\hat{\boldsymbol{\theta}}_i(\boldsymbol{X})} \mathsf{C}_{i,\mathrm{p}}(\hat{\boldsymbol{\theta}}_i(\boldsymbol{X}) \mid \boldsymbol{X}), \tag{2.22}$$

which proves (2.17).

Now we proceed to prove (2.18) as follows. On one hand, note that

$$\begin{aligned}\max_{0\le\Omega\le 1} \min_{\hat{\boldsymbol{\theta}}_0,\hat{\boldsymbol{\theta}}_1} \mathcal{M}(\Omega,\delta_0,\delta_1,\hat{\boldsymbol{\theta}}_0,\hat{\boldsymbol{\theta}}_1) &= \max_{0\le\Omega\le 1} \mathcal{M}(\Omega,\delta_0,\delta_1,\hat{\boldsymbol{\theta}}_0^*,\hat{\boldsymbol{\theta}}_1^*) \\ &\ge \min_{\hat{\boldsymbol{\theta}}_0,\hat{\boldsymbol{\theta}}_1} \max_{0\le\Omega\le 1} \mathcal{M}(\Omega,\delta_0,\delta_1,\hat{\boldsymbol{\theta}}_0,\hat{\boldsymbol{\theta}}_1). \end{aligned} \tag{2.23}$$

On the other hand, we have that for any $\hat{\boldsymbol{\theta}}_0$ and $\hat{\boldsymbol{\theta}}_1$,

$$\max_{0\le\Omega\le 1} \mathcal{M}(\Omega,\delta_0,\delta_1,\hat{\boldsymbol{\theta}}_0,\hat{\boldsymbol{\theta}}_1) \ge \max_{0\le\Omega\le 1} \min_{\hat{\boldsymbol{\theta}}_0,\hat{\boldsymbol{\theta}}_1} \mathcal{M}(\Omega,\delta_0,\delta_1,\hat{\boldsymbol{\theta}}_0,\hat{\boldsymbol{\theta}}_1), \tag{2.24}$$

from which it is clear that

$$\min_{\hat{\boldsymbol{\theta}}_0,\hat{\boldsymbol{\theta}}_1} \max_{0\le\Omega\le 1} \mathcal{M}(\Omega,\delta_0,\delta_1,\hat{\boldsymbol{\theta}}_0,\hat{\boldsymbol{\theta}}_1) \ge \max_{0\le\Omega\le 1} \min_{\hat{\boldsymbol{\theta}}_0,\hat{\boldsymbol{\theta}}_1} \mathcal{M}(\Omega,\delta_0,\delta_1,\hat{\boldsymbol{\theta}}_0,\hat{\boldsymbol{\theta}}_1). \tag{2.25}$$

Combining (2.25) and (2.23), (2.18) is proven.

Moreover, we have

$$\begin{aligned}\min_{\hat{\boldsymbol{\theta}}_0,\hat{\boldsymbol{\theta}}_1} \max_{0\le\Omega\le 1} \mathcal{M}(\Omega,\delta_0,\delta_1,\hat{\boldsymbol{\theta}}_0,\hat{\boldsymbol{\theta}}_1) &= \max_{0\le\Omega\le 1} \mathcal{M}(\Omega,\delta_0,\delta_1,\hat{\boldsymbol{\theta}}_0^*,\hat{\boldsymbol{\theta}}_1^*) \\ &= \max_{i\in\{0,1\}} \left\{ \frac{\int_{\boldsymbol{X}} \delta_i(\boldsymbol{X})\ f_i(\boldsymbol{X})\ \mathsf{C}_{i,\mathrm{p}}^*(\boldsymbol{X})\ d\boldsymbol{X}}{\int_{\boldsymbol{X}} \delta_i(\boldsymbol{X}) f_i(\boldsymbol{X})\ d\boldsymbol{X}} \right\}. \end{aligned} \tag{2.26}$$

■

Theorem 1 reveals that the classical Bayesian estimator is still optimal even when using a subset of the data. Moreover, it shows that regardless of the decision rule, the Bayesian estimator is still optimal. This means that when dividing the joint problem as a detection problem followed by an estimation problem and using a Neyman-Pearson test to decide among the hypotheses in the first step, the Bayesian estimator is optimal for the second step.

2.3.2 Detection

Given the estimators obtained in (2.13) we next determine the optimal detection rules $\delta_0(\boldsymbol{X})$ and $\delta_1(\boldsymbol{X})$. By recalling (2.11) the detection rule is the solution to

$$\mathcal{P}(\alpha,\beta) = \begin{cases} \min_{\{\delta_0,\delta_1\}} & \tilde{\mathcal{L}}(\delta_0,\delta_1), \\ \text{s.t.} & \mathsf{P}_1(\delta_0,\delta_1) \leq \alpha, \\ & \mathsf{P}_2(\delta_0,\delta_1) \leq \beta, \end{cases} \tag{2.27}$$

which is obtained in the following theorem.

Theorem 2 *The problem $\mathcal{P}(\alpha,\beta)$ has a globally optimal solution and the decisions rules $\delta_i(\boldsymbol{X})$ are given by*

$$\delta_1(\boldsymbol{X}) = \begin{cases} 1 \quad \textit{if} & f_0(\boldsymbol{X})\Big[a_1\big(\mathsf{C}^*_{0,\mathrm{p}}(\boldsymbol{X}) - \mathcal{P}(\alpha,\beta)\big) - a_3\Big] \geq \\ & \qquad f_1(\boldsymbol{X})\Big[a_2\big(\mathsf{C}^*_{1,\mathrm{p}}(\boldsymbol{X}) - \mathcal{P}(\alpha,\beta)\big) - a_4\Big], \\ 0 \quad \textit{otherwise}\ , \end{cases} \tag{2.28}$$

where $\{a_i\}$ are non-negative and are selected such that 1) they satisfy $\sum_{i=1}^{4} a_i = 1$ and 2) the detection constraints are satisfied.

Proof: Note that from Theorem 1 we have

$$\begin{aligned}\tilde{\mathcal{L}}(\delta_0,\delta_1) &= \min_{\{\hat{\boldsymbol{\theta}}_0,\hat{\boldsymbol{\theta}}_1\}} \mathcal{L}(\delta_0,\delta_1,\hat{\boldsymbol{\theta}}_0,\hat{\boldsymbol{\theta}}_1) \\ &= \max_{i\in\{0,1\}} \left\{ \frac{\int_{\boldsymbol{X}} \delta_i(\boldsymbol{X})\ f_i(\boldsymbol{X})\ \mathsf{C}^*_{i,\mathrm{p}}(\boldsymbol{X})\ d\boldsymbol{X}}{\int_{\boldsymbol{X}} \delta_i(\boldsymbol{X}) f_i(\boldsymbol{X})\ d\boldsymbol{X}} \right\}. \end{aligned} \tag{2.29}$$

Moreover, from the definitions of $\mathsf{P}_1(\delta_0,\delta_1)$ and $\mathsf{P}_2(\delta_0,\delta_1)$ in (2.8) we have

$$\begin{aligned} \mathsf{P}_1(\delta_0,\delta_1) &= \int_{\boldsymbol{X}} \delta_1(\boldsymbol{X})\ f_0(\boldsymbol{X})\ d\boldsymbol{X} \\ \text{and}\quad \mathsf{P}_2(\delta_0,\delta_1) &= \int_{\boldsymbol{X}} \delta_0(\boldsymbol{X})\ f_1(\boldsymbol{X})\ d\boldsymbol{X}\ . \end{aligned} \tag{2.30}$$

Each term in (2.29) is quasi-linear in $\delta_i(\boldsymbol{X})$, and consequently, quasi-convex[24]. Since taking the weighted maximum preserves quasi-convexity, $\tilde{\mathcal{L}}(\delta_0,\delta_1)$ in (2.29) is quasi-convex, and can be solved by finding the solutions to an equivalent family of feasibility problems [24]. In particular, note that for any given $t \in \mathbb{R}_+$, we have

$$\tilde{\mathcal{L}}(\delta_0,\delta_1) \le t \Leftrightarrow \int_{\boldsymbol{X}} \delta_i(\boldsymbol{X})\ f_i(\boldsymbol{X})\ \left(\mathsf{C}^*_{i,\mathrm{p}}(\boldsymbol{X}) - t\right)\ d\boldsymbol{X} \le 0,\ i=0,1. \tag{2.31}$$

Then, if for a given t, the following feasibility problem is feasible

$$\mathcal{Q}(\alpha,\beta,t) \triangleq \begin{cases} \text{Find} & \delta_0,\delta_1, \\ \text{s.t.} & \int_{\boldsymbol{X}} \delta_0(\boldsymbol{X})\ f_0(\boldsymbol{X})\ \left(\mathsf{C}^*_{0,\mathrm{p}}(\boldsymbol{X}) - t\right)\ d\boldsymbol{X} \le 0, \\ & \int_{\boldsymbol{X}} \delta_1(\boldsymbol{X})\ f_1(\boldsymbol{X})\ \left(\mathsf{C}^*_{1,\mathrm{p}}(\boldsymbol{X}) - t\right)\ d\boldsymbol{X} \le 0, \\ & \int_{\boldsymbol{X}} \delta_1(\boldsymbol{X})\ f_0(\boldsymbol{X})\ d\boldsymbol{X} \le \alpha, \\ & \int_{\boldsymbol{X}} \delta_0(\boldsymbol{X})\ f_1(\boldsymbol{X})\ d\boldsymbol{X} \le \beta, \end{cases} \tag{2.32}$$

then the solution $\mathcal{P}(\alpha,\beta)$ of (2.27) is such that $\mathcal{P}(\alpha,\beta) \leq t$. Conversely, if (2.32) is not feasible, we have $\mathcal{P}(\alpha,\beta) > t$. Given a lower bound $t_{\min}$ and an upper bound $t_{\max}$ known to contain $\mathcal{P}(\alpha,\beta)$, then $\mathcal{P}(\alpha,\beta)$ can be found through a *bi-section* search, solving the *feasibility* problem of (2.32) in each step. Note that (2.32) is equivalent to the following *auxiliary* convex optimization problem

$$\tilde{\mathcal{Q}}(\alpha,\beta,t) \triangleq \begin{cases} \min_{\{\delta_0,\delta_1\}} & \gamma, \\ \text{s.t.} & \int_{\boldsymbol{X}} \delta_0(\boldsymbol{X})\ f_0(\boldsymbol{X})\ \left(\mathsf{C}^*_{0,\mathrm{p}}(\boldsymbol{X}) - t\right)\ d\boldsymbol{X} \leq \gamma, \\ & \int_{\boldsymbol{X}} \delta_1(\boldsymbol{X})\ f_1(\boldsymbol{X})\ \left(\mathsf{C}^*_{1,\mathrm{p}}(\boldsymbol{X}) - t\right)\ d\boldsymbol{X} \leq \gamma, \\ & \int_{\boldsymbol{X}} \delta_1(\boldsymbol{X})\ f_0(\boldsymbol{X})\ d\boldsymbol{X} \leq \alpha + \gamma, \\ & \int_{\boldsymbol{X}} \delta_0(\boldsymbol{X})\ f_1(\boldsymbol{X})\ d\boldsymbol{X} \leq \beta + \gamma, \end{cases} \tag{2.33}$$

in the sense that $\tilde{\mathcal{Q}}(\alpha,\beta,t) \leq 0$ if and only if (2.32) is feasible.

The only remaining part is to solve $\tilde{\mathcal{Q}}(\alpha,\beta,t)$ for any given t. For this purpose, by taking into account the convexity of (2.33), we assign the non-negative Lagrangian multipliers $\boldsymbol{a} \triangleq (a_1,a_2,a_3,a_4)$, that satisfy $\sum_{i=1}^4 a_i = 1$ to the constraints of (2.33) and construct the Lagrange function as

$$\begin{aligned} L(\delta_0,\delta_1,\gamma,\boldsymbol{a}) \ &\triangleq\ (1-\sum_{i=1}^4 a_i)\gamma \\ &+\ a_1 \int_{\boldsymbol{X}} \delta_0(\boldsymbol{X})\ f_0(\boldsymbol{X})\ \left(\mathsf{C}^*_{0,\mathrm{p}}(\boldsymbol{X}) - t\right)\ d\boldsymbol{X} \\ &+\ a_2 \int_{\boldsymbol{X}} \delta_1(\boldsymbol{X})\ f_1(\boldsymbol{X})\ \left(\mathsf{C}^*_{1,\mathrm{p}}(\boldsymbol{X}) - t\right)\ d\boldsymbol{X} \\ &+\ a_3 \int_{\boldsymbol{X}} \delta_1(\boldsymbol{X})\ f_0(\boldsymbol{X})\ d\boldsymbol{X} - a_3\alpha \\ &+\ a_4 \int_{\boldsymbol{X}} \delta_0(\boldsymbol{X})\ f_1(\boldsymbol{X})\ d\boldsymbol{X} - a_4\beta. \end{aligned}$$

Therefore, the Lagrangian dual function is

$$g(\boldsymbol{a}) \stackrel{\triangle}{=} \min_{\delta_0,\delta_1,\gamma} L(\delta_0,\delta_1,\gamma,\boldsymbol{a}) = \min_{\delta_0,\delta_1}(A_0+A_1) - a_3\alpha - a_4\beta$$

where

$$\begin{aligned} A_0 &\stackrel{\triangle}{=} \int_{\boldsymbol{X}} \delta_0(\boldsymbol{X}) \left[a_1 f_0(\boldsymbol{X}) \left(\mathsf{C}^*_{0,\mathrm{p}}(\boldsymbol{X}) - t\right) + a_4 f_1(\boldsymbol{X})\right] d\boldsymbol{X} \ , \\ \text{and} \quad A_1 &\stackrel{\triangle}{=} \int_{\boldsymbol{X}} \delta_1(\boldsymbol{X}) \left[a_2 f_1(\boldsymbol{X}) \left(\mathsf{C}^*_{1,\mathrm{p}}(\boldsymbol{X}) - t\right) + a_3 f_0(\boldsymbol{X})\right] d\boldsymbol{X} \ . \end{aligned}$$

Therefore, the detection rules $\{\delta_0(\boldsymbol{X}), \delta_1(\boldsymbol{X})\}$ that minimize $g(\boldsymbol{a})$ are:

$$\delta_0(\boldsymbol{X}) = 1 \quad \text{if} \quad \begin{cases} a_1 f_0(\boldsymbol{X}) \left(\mathsf{C}^*_{0,\mathrm{p}}(\boldsymbol{X}) - t\right) + a_4 f_1(\boldsymbol{X}) \leq \\ \qquad\qquad a_2 f_1(\boldsymbol{X}) \left(\mathsf{C}^*_{1,\mathrm{p}}(\boldsymbol{X}) - t\right) + a_3 f_0(\boldsymbol{X}) \ , \end{cases}$$

$$\delta_1(\boldsymbol{X}) = 1 \quad \text{if} \quad \begin{cases} a_1 f_0(\boldsymbol{X}) \left(\mathsf{C}^*_{0,\mathrm{p}}(\boldsymbol{X}) - t\right) + a_4 f_1(\boldsymbol{X}) > \\ \qquad\qquad a_2 f_1(\boldsymbol{X}) \left(\mathsf{C}^*_{1,\mathrm{p}}(\boldsymbol{X}) - t\right) + a_3 f_0(\boldsymbol{X}), \end{cases}$$

or in a more compact form as in (2.28). ■

We can find the non-negative multipliers $\boldsymbol{a}$, that satisfy $\|\boldsymbol{a}\|_1 = 1$ and the constraints of (2.33) by performing a numerical search. This can be done by discretizing the interval $[0,1]\times[0,1]\times[0,1]\times[0,1]$ and for each point $\boldsymbol{a}$ in the discretized grid such that $\|\boldsymbol{a}\|_1 = 1$, test whether the resulting decision rules in (2.28) achieve $\tilde{\mathcal{Q}}(\alpha,\beta,t) \leq 0$ in (2.33).

The complete algorithm for finding the optimal detection rule and the associated estimators is summarized in Table 1. This algorithm produces the detection rule (2.28) and the estimators (2.13) associated with each

hypothesis. Then given some observation $\mathbf{X}$, we can compute the test result and the corresponding parameter estimate.

Table 1: The proposed optimal algorithm for joint detection and estimation.

Compute the Neyman-Pearson test with probability of false alarm α, and set the resulting probability of miss detection to be $\beta^*(\alpha)$ and the estimation cost to be $t_{\max}$
if $\beta < \beta^*(\alpha)$, the test is not feasible
  break
else
 Initialize $t_{\min} = 0$
 Evaluate the average posterior costs in (2.3)
 repeat
  $t_0 \leftarrow (t_{\min} + t_{\max})/2$
  $\mathcal{P}(\alpha, \beta) \leftarrow t_0$
  for every $\tilde{\boldsymbol{a}} \succeq 0$ that satisfies $\|\tilde{\boldsymbol{a}}\|_1 = 1$
   Compute the test in (2.28)
   Evaluate $P(\tilde{\boldsymbol{a}}) \triangleq \tilde{\mathcal{Q}}(\alpha, \beta, t)$ of (2.33)
  end for
  if $\min_{\tilde{\boldsymbol{a}}} P(\tilde{\boldsymbol{a}}) \leq 0$
   $t_{\max} \leftarrow t_0$
   $\boldsymbol{a} \leftarrow \arg\min_{\tilde{\boldsymbol{a}}} P(\tilde{\boldsymbol{a}})$
  else
   $t_{\min} \leftarrow t_0$
  end if
 until $t_{\max} - t_{\min} \leq \epsilon$ for ϵ sufficiently small
 $\mathcal{P}(\alpha, \beta) \leftarrow t_{\max}$
 Output the test in (2.28) and estimator of (2.13)
end else

2.3.3 Example: Detection and Estimation with White Gaussian Observations and Unknown Variances

To illustrate the proposed optimal procedure for joint detection and estimation, we consider a simple example of Gaussian observations with unknown variances. Specifically, the composite binary hypothesis test problem is given by

$$\begin{aligned} & \mathsf{H}_0 : \quad \boldsymbol{X} \sim \mathcal{N}(\mathbf{0}, \sigma_0^2 \boldsymbol{I}_N), \\ \text{and} \quad & \mathsf{H}_1 : \quad \boldsymbol{X} \sim \mathcal{N}(\mathbf{0}, \sigma_1^2 \boldsymbol{I}_N), \end{aligned} \tag{2.34}$$

where I_N is the $N \times N$ identity matrix. The parameters to be estimated are the variances σ_0^2 and σ_1^2 under the two hypotheses. An application of this model is in the context of spectrum sensing in cognitive radio systems [25].

For the unknown variances, we assume the following prior distributions

$$\sigma_i^2 \sim \pi_i(\sigma_i^2) = \chi^{-1}(\nu_i, l_i), \quad i = 0, 1, \tag{2.35}$$

where χ^{-1} is a scaled-inverse-chi-squared distribution with parameters ν_i and l_i. We now proceed to find the closed-form expressions for the estimators and the decision rule. For estimating the unknown parameters $\boldsymbol{\theta}_i = \sigma_i^2$, we use the MSE cost as a measure of estimation performance, and therefore, the estimate $\hat{\sigma}_i^2$ is given by the conditional mean $\mathsf{E}_i[\sigma_i^2 \mid \boldsymbol{X}]$ [15]. The

distribution $f_i(\sigma_i^2 \mid \boldsymbol{X})$ is then needed and given by

$$\begin{aligned} f_i(\sigma_i^2 \mid \boldsymbol{X}) &\propto f_i(\boldsymbol{X} \mid \sigma_i^2)\pi_i(\sigma_i^2) && (2.36) \\ &\propto \left(\frac{1}{\sigma_i^N} e^{-\frac{\|\boldsymbol{X}\|_2^2}{2\sigma_i^2}}\right)\left(\frac{1}{\sigma_i^2}^{1+\frac{\nu_i}{2}} e^{-\frac{\nu_i l_i}{2\sigma_i^2}}\right). && (2.37) \end{aligned}$$

That is, $f_i(\sigma_i^2 \mid \boldsymbol{X})$ is a scaled-inverse-chi-squared distribution with parameters $\nu_i + N$ and $\frac{\nu_i l_i + \|\boldsymbol{X}\|_2^2}{\nu_i + N}$. The mean is then given by

$$\hat{\sigma}_i^2 = \frac{\nu_i l_i + \|\boldsymbol{X}\|_2^2}{\nu_i + N - 2}, \tag{2.38}$$

where we have assumed that $\nu_i + N > 2$.

When using the MSE as a measure of the performance cost, it is well-known that the minimum average posterior cost $\mathsf{C}_{i,\mathrm{p}}^*(\boldsymbol{X})$ is given by the posterior variance, that is, $\mathsf{Var}_i[\sigma_i^2 \mid \boldsymbol{X}]$ [15]. Then, knowing that $f_i(\sigma_i^2 \mid \boldsymbol{X})$ is a scaled-inverse-chi-squared distribution, we have

$$\mathsf{C}_{i,\mathrm{p}}^*(\boldsymbol{X}) = \frac{2\left(\nu_i l_i + \|\boldsymbol{X}\|_2^2\right)^2}{(\nu_i + N - 2)^2 (\nu_i + N - 4)}, \tag{2.39}$$

assuming that $\nu_i + N > 4$.

In order to compute the optimal detection rule of Theorem 2 we further

need the distributions $f_i(\boldsymbol{X})$, which is obtain as follows.

$$\begin{aligned} f_i(\boldsymbol{X}) &= \int f_i(\boldsymbol{X} \mid \sigma_i^2)\pi_i(\sigma_i^2)\, d\sigma_i^2 \\ &= \frac{(\nu_i l_i/2)^{\nu_i/2}}{(2\pi)^{N/2}\Gamma(\nu_i/2)} \int \left(\frac{1}{\sigma_i^N} e^{-\frac{\|\boldsymbol{X}\|_2^2}{2\sigma_i^2}}\right)\left(\frac{1}{\sigma_i^2}^{1+\frac{\nu_i}{2}} e^{-\frac{\nu_i l_i}{2\sigma_i^2}}\right) d\sigma_i^2 \\ &= \frac{(\nu_i l_i/2)^{\nu_i/2}}{(2\pi)^{N/2}\Gamma(\nu_i/2)} \frac{\Gamma((\nu_i+N)/2)}{\left(\left(\nu_i l_i + \|\boldsymbol{X}\|_2^2\right)/2\right)^{(\nu_i+N)/2}}, \end{aligned} \tag{2.40}$$

where the last step follows from the fact that the scaled-inverse-chi-squared distribution integrates to one.

Now the steps given in Table 1 can be carried out in order to to obtain the test outcome and the corresponding variance estimate.

In what follows, we present the Neyman-Pearson test and the GLRT for comparison purposes.

Lemma 2 (Neyman-Pearson) *The test that maximizes the detection probability subject to an upper bound on the false alarm probability is*

$$\frac{\left(\nu_0 l_0 + \|\boldsymbol{X}\|_2^2\right)^{(\nu_0+N)/2}}{\left(\nu_1 l_1 + \|\boldsymbol{X}\|_2^2\right)^{(\nu_1+N)/2}} \underset{\mathsf{H}_0}{\overset{\mathsf{H}_1}{\gtrless}} \tilde{\lambda}_{\mathrm{NP}} \ ,$$

where the threshold $\tilde{\lambda}_{\mathrm{NP}}$ is chosen to satisfy the false alarm constraint with equality.

This test needs to be followed by an estimation step. In particular, we can use the MMSE estimate of (2.38).

Theorem 3 *The GLRT for the hypothesis test of (2.34) is given by*

$$\frac{\left(\nu_0 l_0 + \|\boldsymbol{X}\|_2^2\right)^{1+\frac{\nu_0+N}{2}}}{\left(\nu_1 l_1 + \|\boldsymbol{X}\|_2^2\right)^{1+\frac{\nu_1+N}{2}}} \underset{\mathsf{H}_0}{\overset{\mathsf{H}_1}{\gtrless}} \tilde{\lambda}_{\mathrm{GLRT}} \ ;$$

the maximum a posteriori estimate $\tilde{\sigma}_i^2$ *of* σ_i^2 *is given by*

$$\tilde{\sigma}_i^2 = \frac{\nu_i l_i + \|\boldsymbol{X}\|_2^2}{2 + \nu_i + N},$$

and the threshold $\tilde{\lambda}_{\mathrm{GLRT}}$ *is chosen to satisfy the false alarm constraint with equality.*

Proof: The GLRT is given by

$$\frac{\pi_1(\tilde{\sigma}_1^2) f_1(\boldsymbol{X} \mid \tilde{\sigma}_1^2)}{\pi_0(\tilde{\sigma}_0^2) f_0(\boldsymbol{X} \mid \tilde{\sigma}_0^2)} \underset{\mathsf{H}_0}{\overset{\mathsf{H}_1}{\gtrless}} \lambda_{\mathrm{GLRT}} \ ,$$

where the estimate $\tilde{\sigma}_i^2$ is the MAP estimate given by

$$\begin{aligned}
\tilde{\sigma}_i^2 &= \arg\max_{\sigma_i^2} \pi_i(\sigma_i^2) f_i(\boldsymbol{X} \mid \sigma_i^2) \\
&= \arg\max_{\sigma_i^2} \left(\frac{1}{\sigma_i^2}^{1+\frac{\nu_i+N}{2}} e^{-\left(\frac{\nu_i l_i + \|\boldsymbol{X}\|_2^2}{2\sigma_i^2}\right)} \right) \\
&= \frac{\nu_i l_i + \|\boldsymbol{X}\|_2^2}{2 + \nu_i + N}.
\end{aligned}$$

Noticing that $\frac{\nu_i l_i + \|\boldsymbol{X}\|_2^2}{2\tilde{\sigma}_i^2} = \frac{2+\nu_i+N}{2}$ and discarding all terms that do not depend on $\|\boldsymbol{X}\|_2^2$, the test can be rewritten as

$$\frac{\pi_1(\tilde{\sigma}_1^2) f_1(\boldsymbol{X} \mid \tilde{\sigma}_1^2)}{\pi_0(\tilde{\sigma}_0^2) f_0(\boldsymbol{X} \mid \tilde{\sigma}_0^2)} \propto \frac{\left(\nu_0 l_0 + \|\boldsymbol{X}\|_2^2\right)^{1+\frac{\nu_0+N}{2}}}{\left(\nu_1 l_1 + \|\boldsymbol{X}\|_2^2\right)^{1+\frac{\nu_1+N}{2}}} \underset{\mathsf{H}_0}{\overset{\mathsf{H}_1}{\gtrless}} \tilde{\lambda}_{\mathrm{GLRT}}.$$

■

Note that if the prior distributions of the unknown parameters are such that $\nu_0 = \nu_1$, then the Neyman-Pearson test and the GLRT become the same; therefore, both tests achieve the same detection performance. However, the estimation performances are not the same, as the GLRT employs the MAP estimator whereas the Neyman-Pearson method uses the MMSE estimator.

Next we compare the performance of the three methods via simulations. The number of samples is set to be $N = 64$, and the parameters of the prior distributions are set as follows: $\nu_0 = \nu_1 = 10$, $l_0 = 3.2$ and $l_1 = 3.6$. The upper bound on the probability of false alarm for all three tests is set as $\alpha = 0.1$. The resulting probability of miss detection of the Neyman-Pearson test is $\beta^*(\alpha) = 0.52$. As $\nu_0 = \nu_1$, the detection performance of the GLRT is the same as that of the Neyman-Pearson test. Figure 2.1 shows the estimation accuracy as a function of $\Delta P_{miss} \stackrel{\triangle}{=} \beta - \beta^*(\alpha)$. It is seen that as ΔP_{miss} increases, i.e., as the detection performance is allowed to deviate further from the optimal one, the estimation performance monotonically improves. That is, the proposed test trades off between the detection and estimation performances. Moreover, the proposed test outperforms the GLRT in both detection and estimation. Note that the proposed test provides the freedom to work at any point on the curve; that is, we can choose the pair of detection and estimation performances according to the application in hand.

The actual miss detection and false alarm probabilities are shown in Fig. 2.2, where both the miss detection probability $\mathsf{P}_2(\delta_0, \delta_1)$ and the false alarm probability $\mathsf{P}_2(\delta_0.\delta_1)$ are shown as a function of ΔP_{miss}. Interestingly,

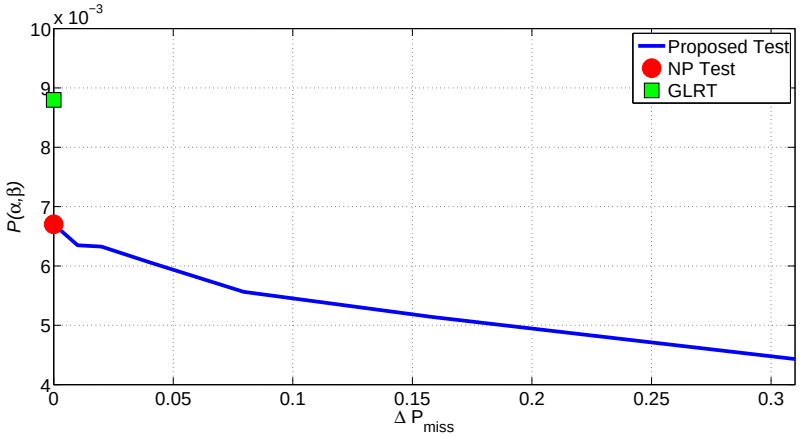


Figure 2.1: The estimation-detection performance tradeoff by the proposed optimal test for the composite hypothesis testing problem in (2.34).

it is seen that the upper bound β imposed on the miss detection probability is always achieved, whereas the gap between the actual false alarm probability and its upper bound α increases with ΔP_{miss}. The exception is when $\Delta P_{miss} = 0$, that is, when the proposed test becomes the Neyman-Pearson test which is known to satisfy both constraints with equalities.

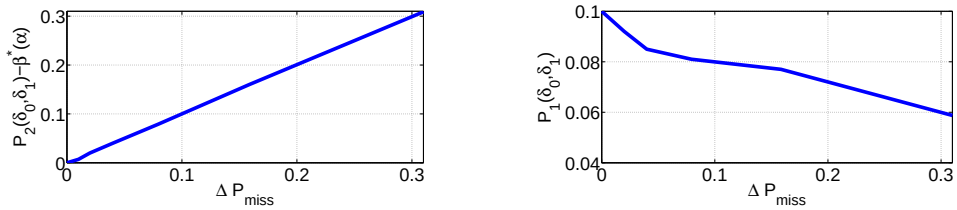


Figure 2.2: The detection performance of the proposed optimal test for the composite hypothesis testing problem in (2.34).

2.3.4 Extension to Multiple Composite Hypothesis Test

In a more general scenario, we wish to decide among M different hypotheses. When observing the signal $\boldsymbol{X}$, we consider the following *composite* hypothesis testing problem:

$$\begin{aligned}
&\mathsf{H}_0: && \boldsymbol{X} \sim f_0(\boldsymbol{X} \mid \boldsymbol{\theta}_0), && \text{with} \quad \boldsymbol{\theta}_0 \sim \pi_0(\boldsymbol{\theta}_0),\\
&\mathsf{H}_1: && \boldsymbol{X} \sim f_1(\boldsymbol{X} \mid \boldsymbol{\theta}_1), && \text{with} \quad \boldsymbol{\theta}_1 \sim \pi_1(\boldsymbol{\theta}_1),\\
&\quad\vdots\\
\text{and} \quad &\mathsf{H}_{M-1}: && \boldsymbol{X} \sim f_{M-1}(\boldsymbol{X} \mid \boldsymbol{\theta}_{M-1}), && \text{with} \quad \boldsymbol{\theta}_{M-1} \sim \pi_{M-1}(\boldsymbol{\theta}_{M-1}).
\end{aligned} \tag{2.41}$$

In this case, we need to find the optimal decision rules $\{\delta_i(\boldsymbol{X})\}$ and estimators $\{\hat{\boldsymbol{\theta}}_i(\boldsymbol{X})\}$ for $i = 0, \ldots, M-1$. We again use an objective function that depends on the estimation performance while assuring that the detection performance satisfies some given constraints. In particular, the objective function is a simple extension of the one used for the binary case, i.e., the average estimation cost of the estimator $\hat{\boldsymbol{\theta}}_i$ when we decide in favor of hypothesis H_i

$$\mathcal{L}_i(\delta_0, \ldots, \delta_{M-1}, \hat{\boldsymbol{\theta}}_i) \triangleq \mathsf{E}_i[\mathsf{C}_i(\hat{\boldsymbol{\theta}}_i(\boldsymbol{X}), \boldsymbol{\theta}_i) \mid \mathsf{D} = \mathsf{H}_i] \ . \tag{2.42}$$

And in order to take into account the estimation performances associated with all hypotheses, we take the maximum

$$\mathcal{L}(\delta_0, \ldots, \delta_{M-1}, \hat{\boldsymbol{\theta}}_0, \ldots, \hat{\boldsymbol{\theta}}_{M-1}) \triangleq \max_{i \in \{0,\ldots,M-1\}} \mathcal{L}_i(\delta_0, \delta_1, \hat{\boldsymbol{\theta}}_i) \ . \tag{2.43}$$

On the other hand, concerning the detection performance, we impose upper bound constraints on the miss detection probabilities of deciding in favor of some other hypothesis when the true hypothesis is H_i, i.e.,

$$\mathsf{P}_i(\delta_0, \ldots \delta_{M-1}) \triangleq \int_{\boldsymbol{X}} (1 - \delta_i(\boldsymbol{X}))\, f_i(\boldsymbol{X})\, d\boldsymbol{X} \leq \omega_i, \qquad i = 0, \ldots, M-1. \tag{2.44}$$

The joint detection and estimation problem for determining the optimal detection rules $\{\delta_0(\boldsymbol{X}), \ldots, \delta_{M-1}(\boldsymbol{X})\}$ and estimators $\{\hat{\boldsymbol{\theta}}_0(\boldsymbol{X}), \ldots, \hat{\boldsymbol{\theta}}_{M-1}(\boldsymbol{X})\}$ is now given by

$$\mathcal{P}(\omega_0, \ldots, \omega_{M-1}) \triangleq \begin{cases} \min_{\{\delta_0, \ldots, \delta_{M-1}, \hat{\boldsymbol{\theta}}_0, \ldots, \hat{\boldsymbol{\theta}}_{M-1}\}} \mathcal{L}(\delta_0, \ldots, \delta_{M-1}, \hat{\boldsymbol{\theta}}_0, \ldots, \hat{\boldsymbol{\theta}}_{M-1}), \\ \text{s.t.} \qquad \mathsf{P}_i(\delta_0, \ldots, \delta_{M-1}) \leq \omega_i, \\ \qquad\qquad \text{for } i = 0, \ldots, M-1, \end{cases} \tag{2.45}$$

with implicit constraints $\sum_{i=0}^{M-1} \delta_i(\boldsymbol{X}) = 1$ and $\delta_0(\boldsymbol{X}), \ldots, \delta_{M-1}(\boldsymbol{X}) \geq 0$.

We proceed by finding the optimal estimators for fixed decision rules, followed by the search for the optimal decision rules. The optimal estimators for fixed detection rules $\{\delta_i(\boldsymbol{X})\}$ are characterized by the following theorem.

Theorem 4 *The solution to the optimization problem*

$$(\hat{\boldsymbol{\theta}}_0^*, \ldots, \hat{\boldsymbol{\theta}}_{M-1}^*) = \arg \min_{\{\hat{\boldsymbol{\theta}}_0, \ldots, \hat{\boldsymbol{\theta}}_{M-1}\}} \mathcal{L}(\delta_0, \ldots, \delta_{M-1}, \hat{\boldsymbol{\theta}}_0, \ldots, \hat{\boldsymbol{\theta}}_{M-1})$$

is

$$\hat{\boldsymbol{\theta}}_i^*(\boldsymbol{X}) \triangleq \arg \inf_{\hat{\boldsymbol{\theta}}_i(\boldsymbol{X})} \mathsf{C}_{i,\mathrm{p}}(\hat{\boldsymbol{\theta}}_i(\boldsymbol{X}) \mid \boldsymbol{X}) \ , \tag{2.46}$$

and

$$\min_{\hat{\boldsymbol{\theta}}_0,\ldots,\hat{\boldsymbol{\theta}}_{M-1}} \mathcal{L}(\delta_0,\ldots,\delta_{M-1},\hat{\boldsymbol{\theta}}_0,\ldots,\hat{\boldsymbol{\theta}}_{M-1}) = \max_{i\in\{0,\ldots,M-1\}} \left\{ \frac{\int_{\boldsymbol{X}} \delta_i(\boldsymbol{X})\; f_i(\boldsymbol{X})\; \mathsf{C}^*_{i,\mathrm{p}}(\boldsymbol{X})\; d\boldsymbol{X}}{\int_{\boldsymbol{X}} \delta_i(\boldsymbol{X}) f_i(\boldsymbol{X})\; d\boldsymbol{X}} \right\}.$$

Proof: The proof follows the same reasoning behind the proof of Theorem 1. ■

Moreover, the optimal detection rules $\delta_0(\boldsymbol{X}),\ldots,\delta_M(\boldsymbol{X})$ are given in the following Theorem.

Theorem 5 *The problem $\mathcal{P}(\omega_0,\ldots,\omega_{M-1})$ has a globally optimal solution and the decision rules $\delta_i(\boldsymbol{X})$ are given by*

$$\delta_i(\boldsymbol{X}) = \begin{cases} 1 \quad \textit{if} \quad f_i(\boldsymbol{X}) \left[a_i^1 \big(\mathsf{C}^*_{i,\mathrm{p}}(\boldsymbol{X}) - \mathcal{P}(\omega_0,\ldots,\omega_M) \big) - a_i^2 \right] \leq \\ \qquad\qquad f_j(\boldsymbol{X}) \left[a_j^1 \big(\mathsf{C}^*_{j,\mathrm{p}}(\boldsymbol{X}) - \mathcal{P}(\omega_0,\ldots,\omega_M) \big) - a_j^2 \right] \\ \qquad\qquad\qquad \textit{for } j = 0,\ldots,M-1 \textit{ and } j \neq i, \\ 0 \qquad \textit{otherwise} \;, \end{cases} \tag{2.47}$$

where $\{a_i^1\}$ and $\{a_i^2\}$ for $i = 0,\ldots,M-1$ are non-negative and are selected such that 1) they satisfy $\sum_{i=1}^{M-1} a_i^1 + \sum_{i=1}^{M-1} a_i^2 = 1$ and 2) the detection constraints are satisfied.

Proof: Note that from Theorem 4 we have

$$\tilde{\mathcal{L}}(\delta_0,\ldots,\delta_{M-1}) = \max_{i\in\{0,\ldots,M-1\}} \left\{ \frac{\int_{\boldsymbol{X}} \delta_i(\boldsymbol{X})\; f_i(\boldsymbol{X})\; \mathsf{C}^*_{i,\mathrm{p}}(\boldsymbol{X})\; d\boldsymbol{X}}{\int_{\boldsymbol{X}} \delta_i(\boldsymbol{X}) f_i(\boldsymbol{X})\; d\boldsymbol{X}} \right\}, \tag{2.48}$$

which needs to be minimized with constraint on the miss detection proba-

bility of each hypothesis, that is,

$$\mathsf{P}_i(\delta_0,\ldots,\delta_{M-1}) = \int_{\boldsymbol{X}} \left(1-\delta_i\left(\boldsymbol{X}\right)\right)\ f_i(\boldsymbol{X})\ d\boldsymbol{X} \quad \text{for } i=0,\ldots,M-1\ . \tag{2.49}$$

Following similar arguments as those in the proof of Theorem 2, we have for $t \in \mathbb{R}_+$

$$\tilde{\mathcal{L}}(\delta_0,\ldots,\delta_{M-1}) \leq t \Leftrightarrow \int_{\boldsymbol{X}} \delta_i(\boldsymbol{X}) f_i(\boldsymbol{X}) \big(\mathsf{C}^*_{i,\mathrm{p}}(\boldsymbol{X}) - t\big) d\boldsymbol{X} \leq 0,\ i=0,\ldots,M-1.$$

Then, if for a given t, the feasibility problem given by

$$\mathcal{Q}(\omega_0,\ldots,\omega_{M-1},t) \triangleq \begin{cases} \text{Find} & \delta_0,\ldots,\delta_{M-1}, \\ \text{s.t.} & \int_{\boldsymbol{X}} \delta_i(\boldsymbol{X})\ f_i(\boldsymbol{X})\ \left(\mathsf{C}^*_{i,\mathrm{p}}(\boldsymbol{X}) - t\right)\ d\boldsymbol{X} \leq 0, \\ & \int_{\boldsymbol{X}} \left(1-\delta_i\left(\boldsymbol{X}\right)\right)\ f_i(\boldsymbol{X})\ d\boldsymbol{X} \leq \omega_i, \\ & \qquad \text{for } i=0,\ldots,M-1 \end{cases} \tag{2.50}$$

is feasible, then the solution $\mathcal{P}(\omega_0,\ldots,\omega_{M-1})$ of (2.27) is such that $\mathcal{P}(\omega_0,\ldots,\omega_{M-1}) \leq t$. Conversely, if (2.50) is not feasible, we have $\mathcal{P}(\omega_0,\ldots,\omega_{M-1}) > t$. The optimal value of $\mathcal{P}(\omega_0,\ldots,\omega_{M-1})$ can be found by a bi-section search on t and for each t solving this feasibility problem.

The feasibility problem of (2.50) can be solved by finding the solution

to the convex optimization problem

$$\tilde{\mathcal{Q}}(\omega_0,\ldots,\omega_{M-1},t)\stackrel{\triangle}{=}\begin{cases}\min_{\{\delta_0,\ldots,\delta_{M-1}\}}\ \gamma,\\ \text{s.t.} \qquad \int_{\boldsymbol{X}}\delta_i(\boldsymbol{X})f_i(\boldsymbol{X})\big(\mathsf{C}^*_{i,\mathrm{p}}(\boldsymbol{X})-t\big)d\boldsymbol{X}\le\gamma,\\ \qquad\qquad \int_{\boldsymbol{X}}\left(1-\delta_i\left(\boldsymbol{X}\right)\right)f_i(\boldsymbol{X})d\boldsymbol{X}\le\omega_i+\gamma,\\ \qquad\qquad\qquad i=0,\ldots,M-1,\end{cases} \tag{2.51}$$

and checking whether $\tilde{\mathcal{Q}}(\omega_0,\ldots,\omega_{M-1},t)\le 0$ or not.

In order to solve $\tilde{\mathcal{Q}}(\omega_0,\ldots,\omega_{M-1},t)$ for any given t, by taking into account the convexity of (2.51), we assign the non-negative Lagrangian multipliers $\boldsymbol{a}^1\stackrel{\triangle}{=}(a_0^1,\ldots,a_{M-1}^1)$ and $\boldsymbol{a}^2\stackrel{\triangle}{=}(a_0^2,\ldots,a_{M-1}^2)$, that satisfy $\|\boldsymbol{a}^1\|_1+\|\boldsymbol{a}^2\|_1=1$ to the constraints of (2.51) and construct the Lagrange function as

$$\begin{aligned}L(\delta_0,\ldots,\delta_{M-1},\gamma,\boldsymbol{a}^1,\boldsymbol{a}^2) \;&\stackrel{\triangle}{=}\; (1-\sum_{i=1}^{M-1}a_i^1-\sum_{i=1}^{M-1}a_i^2)\gamma\\ &+\;\sum_{i=0}^{M-1}a_i^1\int_{\boldsymbol{X}}\delta_i(\boldsymbol{X})\;f_i(\boldsymbol{X})\;\left(\mathsf{C}^*_{i,\mathrm{p}}(\boldsymbol{X})-t\right)\;d\boldsymbol{X}\\ &+\;\sum_{i=0}^{M-1}a_i^2\int_{\boldsymbol{X}}\left(1-\delta_i\left(\boldsymbol{X}\right)\right)\;f_i(\boldsymbol{X})\;d\boldsymbol{X}-a_i^2\omega_i.\end{aligned}$$

The Lagrangian dual function is then given by

$$\begin{aligned}g(\boldsymbol{a}^1,\boldsymbol{a}^2) \;&\stackrel{\triangle}{=}\; \min_{\delta_0,\ldots,\delta_{M-1},\gamma}L(\delta_0,\ldots,\delta_{M-1},\gamma,\boldsymbol{a}^1,\boldsymbol{a}^2)\\ &=\; \min_{\delta_0,\ldots\delta_{M-1}}\sum_{i=0}^{M-1}B_i-\sum_{i=0}^{M-1}a_i^2(1-\omega_i)\end{aligned}$$

where

$$B_i \triangleq \int_{\boldsymbol{X}} \delta_i(\boldsymbol{X})\, f_i(\boldsymbol{X}) \Big[a_i^1 \left(\mathsf{C}^*_{i,\mathrm{p}}(\boldsymbol{X}) - t \right) - a_i^2 \Big]\, d\boldsymbol{X} \ .$$

Therefore, the detection rules $\{\delta_0(\boldsymbol{X}), \ldots, \delta_{M-1}(\boldsymbol{X})\}$ that minimize $g(\boldsymbol{a}^1, \boldsymbol{a}^2)$ are:

$$\delta_i(\boldsymbol{X}) = 1 \quad \text{if} \quad f_i(\boldsymbol{X}) \Big[a_i^1 \big(\mathsf{C}^*_{i,\mathrm{p}}(\boldsymbol{X}) - t \big) - a_i^2 \Big] \leq f_j(\boldsymbol{X}) \left[a_j^1 \big(\mathsf{C}^*_{j,\mathrm{p}}(\boldsymbol{X}) - t \big) - a_j^2 \right]$$
$$j = 0, \ldots, M-1 \text{ and } j \neq i.$$

■

The non-negative multipliers $\boldsymbol{a}^1 = \left[a_0^1, \ldots, a_{M-1}^1\right]^T$ and $\boldsymbol{a}^2 = \left[a_0^2, \ldots, a_{M-1}^2\right]^T$ need to be found using a numerical search, as in the binary case but now in a higher dimensional search space.

2.3.5 Optimal Test with Discrete Observations

In this subsection we consider the special case where the observations take values in a finite discrete set. In this case we can use matrix representations of the different distributions, and each step of the bi-section search corresponds to solving a finite-dimensional linear programming (LP) feasibility problem.

Let $\boldsymbol{X}$ be an observation signal that takes values in a finite discrete set with n possible different realizations. Then, we consider the equivalent observation X with possible values in the set $\{1, \ldots, n\}$. For instance, if we have S observations of a discrete random variable that can take any

of D values, then $n = D^S$. We aim then to solve the *composite* binary hypothesis problem in (2.1) with observation X. Let $P_i(X \mid \boldsymbol{\theta}_i)$ be the known probability mass function (pmf), $\pi_i(\boldsymbol{\theta}_i)$ the known pdf for the parameter $\boldsymbol{\theta}_i$ and $\boldsymbol{p}_i = [p_{1i} \dots p_{ni}]^T$ with $p_{ki} \triangleq P_i(X = k) = \int_{\boldsymbol{\theta}_i} P_i(X = k \mid \boldsymbol{\theta}_i)\pi_i(\boldsymbol{\theta}_i)d\boldsymbol{\theta}_i$ $i \in \{0, 1\}$. We take advantage of the facts that X can only take a finite number of values and that the number of different hypotheses is also finite by representing the randomized test with a $2 \times n$ matrix $\boldsymbol{T} = [t_{ik}]$, whose elements are given by

$$t_{ik} \triangleq P(D = i \mid X = k).$$

Let $\boldsymbol{t}_i$ be the i-th row of $\boldsymbol{T}$. As we always select one of the possible hypotheses, the row vectors needs to satisfy

$$\boldsymbol{t}_0, \boldsymbol{t}_1 \succeq \mathbf{0}, \qquad \text{and} \qquad \boldsymbol{t}_0 + \boldsymbol{t}_1 = \mathbf{1}.$$

Moreover, given the randomized tests $\boldsymbol{t}_0, \boldsymbol{t}_1$, the Type-I and Type-II detection error probabilities are

$$\mathsf{P}_1(\delta_0, \delta_1) = \sum_{k=1}^{n} t_{1k}\, p_{k0} = \boldsymbol{t}_1^T \boldsymbol{p}_0 \qquad \text{and} \qquad \mathsf{P}_2(\delta_0, \delta_1) = \sum_{k=1}^{n} t_{0k}\, p_{k1} = \boldsymbol{t}_0^T \boldsymbol{p}_1. \tag{2.52}$$

Given two non-negative cost functions $\mathsf{C}_0(\hat{\boldsymbol{\theta}}_0, \boldsymbol{\theta}_0)$ and $\mathsf{C}_1(\hat{\boldsymbol{\theta}}_1, \boldsymbol{\theta}_1)$, we want to decide between two hypotheses H_i $(i = 0, 1)$ and provide an estimate for $\boldsymbol{\theta}_i$ only when we decide in favor of H_i. As before, we consider the average estimation cost for estimating $\boldsymbol{\theta}_i$ under H_i when deciding in favor of H_i,

which for a given randomized policy $\{\boldsymbol{t}_0, \boldsymbol{t}_1\}$ and estimator $\hat{\boldsymbol{\theta}}_i(X)$ is given by

$$\mathcal{L}_i(\boldsymbol{t}_0, \boldsymbol{t}_1, \hat{\boldsymbol{\theta}}_i) \triangleq \mathsf{E}_i[\mathsf{C}_i(\hat{\boldsymbol{\theta}}_i(X), \boldsymbol{\theta}_i) \mid \mathsf{D} = \mathsf{H}_i], \quad i = 0, 1, \tag{2.53}$$

and use the maximum of these two average estimation costs as the overall performance measure, i.e.,

$$\mathcal{L}_i(\boldsymbol{t}_0, \boldsymbol{t}_1, \hat{\boldsymbol{\theta}}_0, \hat{\boldsymbol{\theta}}_1) \triangleq \max_{i=\{0,1\}} \mathcal{L}_i(\boldsymbol{t}_0, \boldsymbol{t}_1, \hat{\boldsymbol{\theta}}_i). \tag{2.54}$$

The combined problem for determining the optimal decision rules $\{\boldsymbol{t}_0, \boldsymbol{t}_1\}$ and estimators
$\{\hat{\boldsymbol{\theta}}_0(X), \hat{\boldsymbol{\theta}}_1(X)\}$ is then

$$\mathcal{P}(\alpha, \beta) = \begin{cases} \min_{\{\boldsymbol{t}_0, \boldsymbol{t}_1, \hat{\boldsymbol{\theta}}_0, \hat{\boldsymbol{\theta}}_1\}}, & \mathcal{L}(\boldsymbol{t}_0, \boldsymbol{t}_1, \hat{\boldsymbol{\theta}}_0, \hat{\boldsymbol{\theta}}_1), \\ \text{s.t.} & \mathsf{P}_1(\delta_0, \delta_1) \leq \alpha, \\ & \mathsf{P}_2(\delta_0, \delta_1) \leq \beta, \end{cases} \tag{2.55}$$

where we also have the implicit constraints $\boldsymbol{t}_0(X) + \boldsymbol{t}_1(X) = \mathbf{1}$ and $\boldsymbol{t}_0(X), \boldsymbol{t}_1(X) \succeq 0$.

The optimal composite test is characterized by the following two Theorems.

Theorem 6 *The solution to the optimization problem*

$$(\hat{\boldsymbol{\theta}}_0^*, \hat{\boldsymbol{\theta}}_1^*) = \arg \min_{\{\hat{\boldsymbol{\theta}}_0, \hat{\boldsymbol{\theta}}_1\}} \mathcal{L}(\boldsymbol{t}_0, \boldsymbol{t}_1, \hat{\boldsymbol{\theta}}_0, \hat{\boldsymbol{\theta}}_1)$$

is

$$\hat{\boldsymbol{\theta}}_i^*(X) \triangleq \arg \inf_{\hat{\boldsymbol{\theta}}_i(X)} \mathsf{C}_{i,\mathrm{p}}(\hat{\boldsymbol{\theta}}_i(X) \mid X). \tag{2.56}$$

Proof: This proof is similar to that of Theorem 1. When finding the lower bounds in (2.21), in the case of discrete observations we have, for $i \in \{0, 1\}$,

$$\begin{aligned}
&\mathsf{E}_i[P(D = i \mid X)\mathsf{C}_i(\hat{\boldsymbol{\theta}}_i(X), \boldsymbol{\theta}_i)] \\
&\quad = \sum_{k=1}^{n} \int_{\boldsymbol{\theta}_i} t_{ik} \mathsf{C}_i(\hat{\boldsymbol{\theta}}_i(X = k), \boldsymbol{\theta}_i) P_i(X = k \mid \boldsymbol{\theta}_i) \pi_i(\boldsymbol{\theta}_i) d\boldsymbol{\theta}_i \\
&\quad = \sum_{k=1}^{n} t_{ik} \int_{\boldsymbol{\theta}_i} \Big(\mathsf{C}_i(\hat{\boldsymbol{\theta}}_i(X = k), \boldsymbol{\theta}_i) P_i(X = k \mid \boldsymbol{\theta}_i) \pi_i(\boldsymbol{\theta}_i) d\boldsymbol{\theta}_i \Big) \\
&\quad = \sum_{k=1}^{n} t_{ik} P_i(X = k) \int_{\boldsymbol{\theta}_i} \Big(\mathsf{C}_i(\hat{\boldsymbol{\theta}}_i(X = k), \boldsymbol{\theta}_i) \pi_i(\boldsymbol{\theta}_i \mid X = k) d\boldsymbol{\theta}_i \Big) \\
&\quad = \sum_{k=1}^{n} t_{ik} P_i(X = k) \mathsf{E}_{\boldsymbol{\theta}_i}[\mathsf{C}(\hat{\boldsymbol{\theta}}_i(X = k), \boldsymbol{\theta}_i) \mid X = k] \\
&\quad = \sum_{k=1}^{n} t_{ik} P_i(X = k) \mathsf{C}_{i,\mathrm{p}}(\hat{\boldsymbol{\theta}}_i(X = k) \mid X = k) \\
&\quad \geq \sum_{k=1}^{n} t_{ik} P_i(X = k) \inf_{\boldsymbol{U}} \mathsf{C}_{i,\mathrm{p}}(\boldsymbol{U} \mid X = k) \\
&\quad = \sum_{k=1}^{n} t_{ik} P_i(X = k) \mathsf{C}_{i,\mathrm{p}}^*(X = k) \\
&\quad = \boldsymbol{t}_i^T \boldsymbol{C}_i \boldsymbol{p}_i \ ,
\end{aligned} \tag{2.57}$$

where $\boldsymbol{C}_i$ is an $n \times n$ diagonal matrix with the k-th diagonal entry equal to $\mathsf{C}_{i,\mathrm{p}}^*(X = k)$. These lower bounds can be achieved by setting the estimators as

$$\hat{\boldsymbol{\theta}}_i^*(X) \triangleq \arg \inf_{U} \mathsf{C}_{i,\mathrm{p}}(U \mid X). \tag{2.58}$$

Moreover, we obtain

$$\mathcal{L}(\boldsymbol{t}_0, \boldsymbol{t}_1, \hat{\boldsymbol{\theta}}_0^*, \hat{\boldsymbol{\theta}}_1^*) = \max_{i \in \{0,1\}} \left\{ \frac{\boldsymbol{t}_i^T \boldsymbol{C}_i \boldsymbol{p}_i}{\boldsymbol{t}_i^T \boldsymbol{p}_i} \right\}. \tag{2.59}$$

■

Theorem 7 *The test that solves the detection problem $\mathcal{P}(\alpha, \beta)$ is given by the solution to the following optimization problem*

$$\begin{array}{ll} \min_{\{\boldsymbol{t}_0, \boldsymbol{t}_1\}} & \gamma \\ \text{s.t.} & \boldsymbol{t}_0^T \big(\boldsymbol{C}_0 - \mathcal{P}(\alpha, \beta) \boldsymbol{I}_n\big) \boldsymbol{p}_0 \leq \gamma \\ & \boldsymbol{t}_1^T \big(\boldsymbol{C}_1 - \mathcal{P}(\alpha, \beta) \boldsymbol{I}_n\big) \boldsymbol{p}_1 \leq \gamma \\ & \boldsymbol{t}_1^T \boldsymbol{p}_0 \leq \alpha + \gamma \\ & \boldsymbol{t}_0^T \boldsymbol{p}_1 \leq \beta + \gamma \\ & \boldsymbol{t}_0 \succeq \boldsymbol{0}, \quad \boldsymbol{t}_1 \succeq \boldsymbol{0}, \quad \boldsymbol{t}_0 + \boldsymbol{t}_1 = \boldsymbol{1}, \end{array} \tag{2.60}$$

where $\boldsymbol{I}_n$ is the $n \times n$ identity matrix and $\boldsymbol{C}_i$ is an $n \times n$ diagonal matrix with its k-th diagonal entry as $\mathsf{C}_{i,\mathrm{p}}^*(X = k)$.

Proof: The proof is similar to that of Theorem 2. For a given $t \in \mathbb{R}_+$ and noticing that $\mathcal{L}(\boldsymbol{t}_0, \boldsymbol{t}_1, \hat{\boldsymbol{\theta}}_0^*, \hat{\boldsymbol{\theta}}_1^*) \leq t \iff \boldsymbol{t}_i^T \left(\boldsymbol{C}_i - t \boldsymbol{I}_n\right) \boldsymbol{p}_i \leq 0$, $i = 0, 1$, we

then need to solve the following feasibility problem

$$\mathcal{Q}(\alpha,\beta,t) \triangleq \begin{cases} \text{Find} & \boldsymbol{t}_0, \boldsymbol{t}_1, \\ \text{s.t.} & \boldsymbol{t}_0^T \left(\boldsymbol{C}_0 - t\boldsymbol{I}_n\right) \boldsymbol{p}_0 \leq 0, \\ & \boldsymbol{t}_1^T \left(\boldsymbol{C}_1 - t\boldsymbol{I}_n\right) \boldsymbol{p}_1 \leq 0, \\ & \boldsymbol{t}_1^T \boldsymbol{p}_0 \leq \alpha, \\ & \boldsymbol{t}_0^T \boldsymbol{p}_1 \leq \beta, \\ & \boldsymbol{t}_0 \succeq \boldsymbol{0}, \quad \boldsymbol{t}_1 \succeq \boldsymbol{0}, \quad \boldsymbol{t}_0 + \boldsymbol{t}_1 = \boldsymbol{1}, \end{cases} \tag{2.61}$$

which is feasible if and only if the following *auxiliary* convex optimization problem has a negative solution:

$$\tilde{\mathcal{Q}}(\alpha,\beta,t) = \begin{cases} \min_{\{\boldsymbol{t}_0,\boldsymbol{t}_1\}} & \gamma, \\ \text{s.t.} & \boldsymbol{t}_0^T \left(\boldsymbol{C}_0 - t\boldsymbol{I}_n\right) \boldsymbol{p}_0 \leq \gamma, \\ & \boldsymbol{t}_1^T \left(\boldsymbol{C}_1 - t\boldsymbol{I}_n\right) \boldsymbol{p}_1 \leq \gamma, \\ & \boldsymbol{t}_1^T \boldsymbol{p}_0 \leq \alpha + \gamma, \\ & \boldsymbol{t}_0^T \boldsymbol{p}_1 \leq \beta + \gamma, \\ & \boldsymbol{t}_0 \succeq \boldsymbol{0}, \quad \boldsymbol{t}_1 \succeq \boldsymbol{0}, \quad \boldsymbol{t}_0 + \boldsymbol{t}_1 = \boldsymbol{1}. \end{cases} \tag{2.62}$$

That is, $\mathcal{Q}(\alpha,\beta,t)$ is feasible if and only if $\tilde{\mathcal{Q}}(\alpha,\beta,t) \leq 0$. ■

Notice that problem (2.62) is a linear programming (LP) problem, and therefore, it can be solved using a standard LP solver.

The test can be carried out by following the steps in Table 1, replacing steps $10-13$ with the solution of the LP in (2.62). That is, the numerical search of multipliers is replaced by solving an LP. If the number of possible realizations n of X is such that solving the LP is numerically more demand-

ing than the search of the multipliers, then the optimal test with discrete observations can be found by replacing (2.13) and (2.28) in Table 1 with (2.56) and

$$t_{1k} = \begin{cases} 1 \quad \text{if} & p_{k0}\Big[a_1\big(\mathsf{C}^*_{0,\mathrm{p}}(X=k) - \mathcal{P}(\alpha,\beta)\big) - a_3\Big] \geq \\ & \qquad p_{k1}\Big[a_2\big(\mathsf{C}^*_{1,\mathrm{p}}(X=k) - \mathcal{P}(\alpha,\beta)\big) - a_4\Big] \,, \\ 0 \quad \text{otherwise} \,, \end{cases} \tag{2.63}$$

respectively.

2.4 Optimal Detection and Estimation of Periodicities in DNA Sequences

2.4.1 Background

DNA sequences present numerous types of regularities and repetitions that need to be detected and estimated in order to discover the underlying structures and properties. For example, periodicities of various lengths and various types have been shown to be related to the evolution of the genome and protein structure [26, 27]. In particular, a periodicity of 21 bases is linked with the α-helix formation protein molecules [27] and a periodicity of three is associated with the protein coding regions of the DNA.

A DNA sequence is the concatenation of nucleotides. There are four different nucleotides that are the basic units of DNA: Adenine (A), Cytosine (C), Guanine (G) and Thymine (T), each with different biochemical properties. Possible periodicities in these sequences are classified as homologous, eroded and latent [28]. Homologous or perfect periodicities consist

of a segment of DNA that is repeated periodically in the sequence. On the other hand, in case of eroded or imperfect periodicities, the repeating segment exhibits mutations in nucleotides at certain positions. In the case of latent periodicities, the repeating segment is not fixed but only has some specific constraints. For example, the latent periodicity $(A/G)T(G/C/T)$ refers to a periodicity of three nucleotides where both the nucleotide A and nucleotide G are found in the first position of the segment most of the time the segment is observed; the nucleotide T is most likely found in the second position; and last position can be either G, C or T.

Methods for detecting and estimating periodicities in sequences can be classified in two categories [29], i.e., exploratory and confirmatory. The former is designed to discover the main periodicity component in a sequence, while the latter seeks to determine the strength of this component. Moreover, some methods map the symbolic sequence to a numeric one and then process the numeric sequence; whereas others operate directly on the symbolic sequence. An example of an exploratory approach that requires a suitable symbolic-to-numeric mapping is the Fourier-based method. The symbolic sequence is first converted to a discrete-time signal by mapping each nucleotide to a number. The discrete Fourier transform (DFT) is then applied to the signal to obtain its spectrum from which periodic components can be identified [30]. Note that mapping nucleotides to numbers introduces an artificial structure that is not inherent to the original DNA sequence.

In [31] and [32] a method is proposed to find periodic components in a sequence from a *pure* estimation perspective of the problem within a statistical model. In particular, periodicities are inferred based on the maximum-

likelihood estimates of the statistical distribution of the repeating segment and the period. In what follows, we adopt the statistical model in [31, 32] and restate the problem as a joint detection and estimation one. That is, we would like to determine if a sequence posses a periodic component, and if so, to estimate its period. We then apply the proposed test to solve the problem.

2.4.2 Problem Formulation

A biomolecular sequence is defined as a length-N sequence, denoted as $\boldsymbol{X} = (x_1 \dots x_N)$, where each element x_i belongs to a finite alphabet $\mathcal{A}$. In the case of DNA, $\mathcal{A} \triangleq \{A, C, G, T\}$, representing the nucleotides Adenine, Cytosine, Guanine and Thymine respectively.

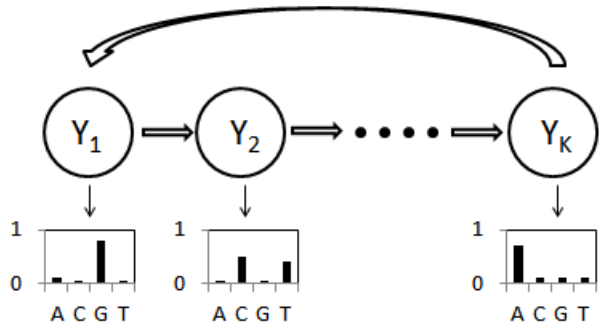


Figure 2.3: A hidden Markov model for a DNA sequence with periodicity of K nucleotides.

Each nucleotide of a DNA sequence with a periodicity of K nucleotides is modeled as a realization of an information source with some underlying probability mass function. The sequence is generated by cyclicly drawing symbols from K such sources as shown in Fig. 2.3. This can be represented

as a first-order hidden Markov model with K states: $Y_1, Y_2, \ldots Y_K$. The transition probability from a state to the following one is 1, i.e., Y_i to Y_{i+1} for $i = 1, \ldots, K-1$ and from the last state to the first one, i.e., Y_K to Y_1, is one. Each state has an emission probability described by $\boldsymbol{p}_i = [p_i(1) \ldots p_i(4)]^T$, where $p_i(j) \triangleq P(Y_i = a_j)$, $a_j \in \mathcal{A}$. These distributions can be combined to form the position weight matrix $\boldsymbol{Q}^K = [\boldsymbol{p}_1 \ \ldots \boldsymbol{p}_K]$ which is unknown. We are interested in estimating only the period K, but not the position weight matrix $\boldsymbol{Q}^K$. In [32], both $\boldsymbol{Q}^K$ and K are estimated. In our approach, using a prior distribution for $\boldsymbol{Q}^K$, we integrate out this nuisance parameter. Therefore, we define $\boldsymbol{\theta}_1 \triangleq K$.

On the other hand, when an observed DNA sequence does not posses periodicity, it is assumed that each nucleotide in the sequence is a realization of a random variable that follows a background distribution $\boldsymbol{q} = [q(1) \ldots q(4)]^T$, where $q(j) \triangleq P(Y = a_j)$, $a_j \in \mathcal{A}$. When deciding that there is no periodicity present, we need to estimate the background distribution as it describes the structure of the sequence. Hence, $\boldsymbol{\theta}_0 = \boldsymbol{q}$.

In summary, we aim to detect whether an observed DNA sequence $\boldsymbol{X}$ has an underlying periodicity and if so, estimate the period K. On the other hand, when we decide that the observed sequence does not have an underlying periodicity, we estimate the background distribution $\boldsymbol{q}$ of the nucleotides. Next we derive the optimal test for the above DNA periodicity detection/estimation problem using the theory developed in Section 2.3.

2.4.3 The Jointly Optimal Test

In this subsection we derive the distributions that are needed to carry out the optimal test of (2.28). We will obtain the estimators and the associated costs under H_1 and H_0 respectively.

2.4.3.1 Periodic DNA sequences

The number of complete periods that are observed in a sequence $\boldsymbol{X}$ of length N and periodicity K is $M \stackrel{\triangle}{=} \lfloor \frac{N}{K} \rfloor$, with $\lfloor . \rfloor$ being the flooring operator. As we need to know the index of each nucleotide within a period, we define $\bar{i} \stackrel{\triangle}{=} 1 + ((i-1) \mod K)$.

The likelihood of observing a sequence with periodicity K is given by

$$\begin{aligned} f_1(\boldsymbol{X} \mid K, \boldsymbol{Q}^K) &= \prod_{i=1}^{N} P(Y_{\bar{i}} = x_i \mid K, \boldsymbol{Q}^K) \\ &= \prod_{i=1}^{N} \boldsymbol{p}_{\bar{i}}^{\boldsymbol{n}(x_i)} = \prod_{j=1}^{K} \boldsymbol{p}_j^{\sum_{i=0}^{M-1} \boldsymbol{n}(x_{K(i-1)+j})}, \end{aligned}$$

where $\boldsymbol{n}(x_i)$ is a four-dimensional vector with a one in the j-th position if $x_i = a_j$ and zeroes elsewhere; and we denote $\boldsymbol{a}^{\boldsymbol{c}} \stackrel{\triangle}{=} \prod_{j=1}^{4} a(j)^{c(j)}$ for vectors $\boldsymbol{a} = [a(1) \ldots a(4)]^T$ and $\boldsymbol{c} = [c(1) \ldots c(4)]^T$.

We assume that $\boldsymbol{p}_1, \boldsymbol{p}_2, \ldots \boldsymbol{p}_K$ are independent, each with a Dirichlet distribution with parameters $\boldsymbol{\alpha}_k \stackrel{\triangle}{=} [\alpha_k(1) \ldots \alpha_k(4)]^T$, i.e.,

$$\pi_1(\boldsymbol{Q}^K) = \prod_{k=1}^{K} \frac{1}{B(\boldsymbol{\alpha}_k)} \prod_{j=1}^{4} p_k(j)^{\alpha_k(j)-1} = \prod_{k=1}^{K} \frac{1}{B(\boldsymbol{\alpha}_k)} \boldsymbol{p}_k^{\boldsymbol{\alpha}_k - \mathbf{1}},$$

where

$$B(\boldsymbol{\alpha}_k) \triangleq \frac{\prod_{j=1}^{4}\Gamma(\alpha_k(j))}{\Gamma(\sum_{j=1}^{4}\alpha_i(j))},$$

Γ is the gamma function and $\mathbf{1} = [1,\ldots,1]^T$. Then, a closed-form expression for $f_1(\boldsymbol{X} \mid K)$ can be obtained by integrating out $\boldsymbol{p}_1, \boldsymbol{p}_2, \ldots \boldsymbol{p}_K$ as follows:

$$\begin{aligned} f_1(\boldsymbol{X} \mid K) &= \int f_1(\boldsymbol{X} \mid K, \boldsymbol{Q}^K)\pi_1(\boldsymbol{Q}^K)\, d\boldsymbol{Q}^K \\ &= \prod_{j=1}^{K}\int \frac{1}{B(\boldsymbol{\alpha}_j)}\boldsymbol{p}_j^{\sum_{i=0}^{M-1}\boldsymbol{n}(x_{K(i-1)+j})+\boldsymbol{\alpha}_j-\mathbf{1}}\, d\boldsymbol{p}_j \\ &= \prod_{j=1}^{K}\frac{B\left(\sum_{i=0}^{M-1}\boldsymbol{n}(x_{K(i-1)+j})+\boldsymbol{\alpha}_j\right)}{B(\boldsymbol{\alpha}_j)}. \end{aligned} \tag{2.64}$$

Using (2.64) and assuming a uniform prior $\pi_1(k)$ on K between K_l and K_u, the marginal distribution $f_1(\boldsymbol{X})$ can be obtained as

$$\begin{aligned} f_1(\boldsymbol{X}) &= \sum_{k=K_l}^{K_u} f_1(\boldsymbol{X} \mid k)\pi_1(k) \\ &= \frac{1}{K_u - K_l}\sum_{k=K_l}^{K_u}\prod_{j=1}^{k}\frac{B\left(\sum_{i=0}^{M-1}\boldsymbol{n}(x_{k(i-1)+j})+\boldsymbol{\alpha}_j\right)}{B(\boldsymbol{\alpha}_j)}, \end{aligned} \tag{2.65}$$

which can be computed numerically.

The posterior of the unknown periodicity is then given by

$$\begin{aligned} f_1(K = k \mid \boldsymbol{X}) &= \frac{f_1(\boldsymbol{X} \mid k)\pi_1(k)}{f_1(\boldsymbol{X})} \\ &= \frac{\prod_{j=1}^{k}\frac{B\left(\sum_{i=0}^{M-1}\boldsymbol{n}(x_{k(i-1)+j})+\boldsymbol{\alpha}_j\right)}{B(\boldsymbol{\alpha}_j)}}{\sum_{k=K_l}^{K_u} f_1(\boldsymbol{X} \mid k)}. \end{aligned} \tag{2.66}$$

Hence, the MMSE estimate of the period is given by

$$\begin{aligned}\hat{K} = \mathsf{E}_1(K \mid \boldsymbol{X}) &= \sum_{k=K_l}^{K_u} k f_1(K=k \mid \boldsymbol{X}) \\ &= \frac{\sum_{k=K_l}^{K_u} k \prod_{j=1}^{k} \frac{B\left(\sum_{i=0}^{M-1} \boldsymbol{n}(x_{k(i-1)+j}) + \boldsymbol{\alpha}_j\right)}{B(\boldsymbol{\alpha}_j)}}{\sum_{k=K_l}^{K_u} f_1(\boldsymbol{X} \mid k)}. \quad (2.67)\end{aligned}$$

The average posterior cost $\mathsf{C}^*_{1,\mathrm{p}}(\boldsymbol{X})$ becomes

$$\begin{aligned}\mathsf{C}^*_{1,\mathrm{p}}(\boldsymbol{X}) &= \mathsf{Var}_1(K \mid \boldsymbol{X}) \\ &= \sum_{k=K_l}^{K_u} \left(k - \hat{K}\right)^2 f_1(K=k \mid \boldsymbol{X}) \\ &= \frac{\sum_{k=K_l}^{K_u} \left(k - \hat{\boldsymbol{\theta}}_1\right)^2 \prod_{j=1}^{k} \frac{B\left(\sum_{i=0}^{M-1} \boldsymbol{n}(x_{k(i-1)+j}) + \boldsymbol{\alpha}_j\right)}{B(\boldsymbol{\alpha}_j)}}{\sum_{k=K_l}^{K_u} f_1(\boldsymbol{X} \mid k)}. \quad (2.68)\end{aligned}$$

2.4.3.2 Aperiodic DNA sequences

For sequences with no periodicity, each nucleotide is independent and identically distributed according to a background distribution $\boldsymbol{q}$. The unknown parameter in this case is $\boldsymbol{\theta}_0 = \boldsymbol{q}$.

Given $\boldsymbol{q}$, the likelihood of the observation is

$$f_0(\boldsymbol{X} \mid \boldsymbol{q}) = \prod_{i=1}^{N} P(Y = x_i \mid \boldsymbol{q}) = \prod_{i=1}^{N} \boldsymbol{q}^{\boldsymbol{n}(x_i)} = \boldsymbol{q}^{\sum_{i=0}^{N-1} \boldsymbol{n}(x_i)},$$

which can be seen as a special case of the periodic sequence with a period of 1 nucleotide. Assuming a Dirichlet prior on $\boldsymbol{q}$ with known parameters

$\boldsymbol{\beta} = [\beta(1) \ldots \beta(4)]^T$; i.e.,

$$\pi_0(\boldsymbol{q}) = \frac{1}{B(\boldsymbol{\beta})} \boldsymbol{q}^{\boldsymbol{\beta}-\mathbf{1}},$$

then

$$\begin{aligned} f_0(\boldsymbol{X}) &= \int f_0(\boldsymbol{X} \mid \boldsymbol{q}) \pi_0(\boldsymbol{q}) \, d\boldsymbol{q} \\ &= \frac{B\left(\sum_{i=0}^{N-1} \boldsymbol{n}(x_i) + \boldsymbol{\beta}\right)}{B(\boldsymbol{\beta})}. \end{aligned}$$

Furthermore, we have

$$\begin{aligned} f_0(\boldsymbol{q} \mid \boldsymbol{X}) &\propto f_0(\boldsymbol{X} \mid \boldsymbol{q}) \pi_0(\boldsymbol{q}) \\ &= \left(\prod_{i=1}^{N} \boldsymbol{q}^{\boldsymbol{n}(x_i)} \right) \frac{1}{B(\boldsymbol{\beta})} \boldsymbol{p}^{\boldsymbol{\beta}-\mathbf{1}} \\ &= \frac{1}{B(\boldsymbol{\beta})} \boldsymbol{q}^{\sum_{i=0}^{N-1} \boldsymbol{n}(x_i) + \boldsymbol{\beta} - \mathbf{1}}, \end{aligned} \tag{2.69}$$

that is, the posterior is also a Dirichlet distribution. Then, the MMSE estimate of $\boldsymbol{q}$ is given by

$$\hat{\boldsymbol{q}} = \mathsf{E}_0[\boldsymbol{q} \mid \boldsymbol{X}] = \frac{\sum_{i=0}^{N-1} \boldsymbol{n}(x_i) + \boldsymbol{\beta}}{N + \mathbf{1}^T \boldsymbol{\beta}}.$$

Moreover, the cost under H_0 is given by

$$\begin{aligned}\mathsf{C}^*_{0,\mathrm{p}}(\boldsymbol{X}) &= \sum_{j=1}^{4} \mathsf{Var}_0[q(j) \mid \boldsymbol{X}] \\ &= \left\| \frac{\left(\sum_{i=0}^{N-1} \boldsymbol{n}(x_i) + \boldsymbol{\beta}\right) \odot \left(N\mathbf{1} + \mathbf{1}^T\boldsymbol{\beta}\mathbf{1} - \sum_{i=0}^{N-1} \boldsymbol{n}(x_i) - \boldsymbol{\beta}\right)}{\left(N + \mathbf{1}^T\boldsymbol{\beta}\right)^2 \left(N + \mathbf{1}^T\boldsymbol{\beta} + 1\right)} \right\|_1,\end{aligned}$$

where $\odot$ is the componentwise multiplication of vectors, that is, $[a_1, \ldots, a_4]^T \odot [b_1, \ldots, b_4]^T \triangleq [a_1 b_1, \ldots, a_4 b_4]^T$.

Clearly, a DNA sequence can only take values on a finite discrete space, i.e., there are $n = 4^N$ possible different length-N sequences. However, for moderate values of N, n is such that searching for 4 multipliers is numerically more efficient that solving the LP of (2.62). Therefore, we find the optimal test using (2.13) and (2.28), that is, following the steps given in Table 1.

2.4.4 Simulation Results

For the simulations, we consider DNA sequences of length 100, which are considered as short sequences. The bound on the probability of false alarm is set to be $\alpha = 0.001$. The parameters of the priors are set as follows. For aperiodic sequences, the parameters of the prior distribution are $\boldsymbol{\beta} = 2\ \mathbf{1}$. For periodic sequences, we set $K_l = 2$ and $K_u = 10$. The parameters $\boldsymbol{\alpha}_k$, $k = 1, \ldots, K$, of the prior for the position weight matrix $\boldsymbol{Q}^K$ are columns from 1 to K of the matrix

$$\boldsymbol{\alpha}_{1:K_u} = \begin{bmatrix} 4.5 & 5.4 & 8.0 & 6.6 & 5.2 & 3.8 & 5.4 & 4.4 & 3.9 & 8.8 \\ 4.2 & 9.8 & 5.9 & 6.2 & 9.8 & 9.2 & 8.1 & 11.2 & 4.9 & 7.5 \\ 8.7 & 8.8 & 11.2 & 6.6 & 6.7 & 6.7 & 3.9 & 4.7 & 2.9 & 6.3 \\ 10.4 & 2.1 & 2.0 & 9.7 & 2.4 & 3.5 & 9.4 & 9.7 & 7.8 & 8.4 \end{bmatrix}. \quad (2.70)$$

We first compute the probability of miss detection under the Neyman-Pearson test, which is $\beta^*(\alpha) = 0.31$. The resulting estimation costs for different values of $\Delta P_{miss} \stackrel{\triangle}{=} \beta - \beta^*(\alpha)$ are shown in Fig. 2.4. The performance of the GLRT is also shown. It is seen that the proposed test outperforms the GLRT for a given ΔP_{miss}. Moreover, the tradeoff between the detection and estimation performances of the proposed test is clearly shown and we have the flexibility of operating on any point of the tradeoff curve. The detection performance is shown in Fig. 2.5, where it is seen that the constraint on the miss detection is attained with equality whereas the constraint on the false alarm probability is achieved with equality only for $\Delta P_{miss} = 0$.

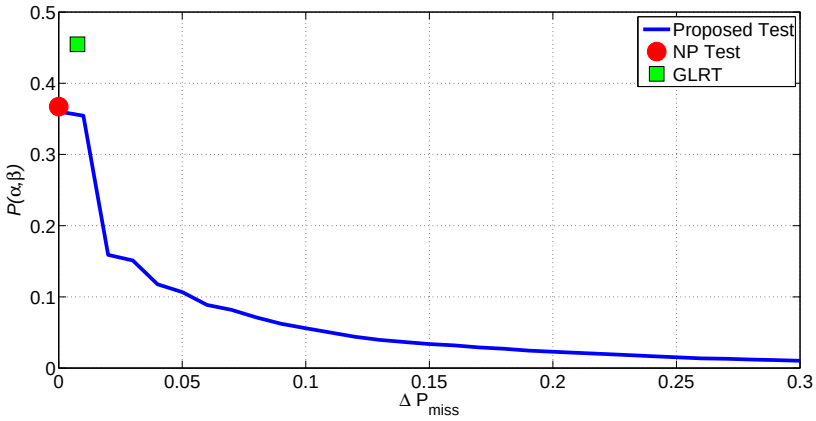


Figure 2.4: The estimation-detection performance tradeoff for DNA periodicity detection and estimation.

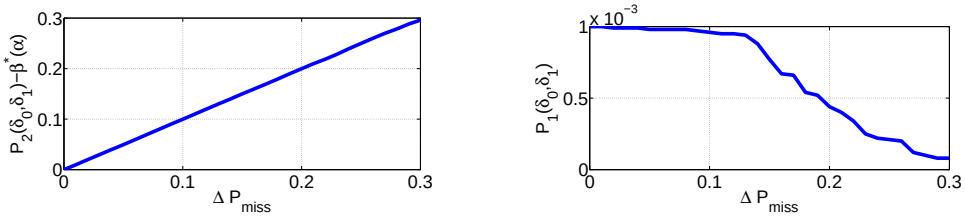


Figure 2.5: The detection performance for DNA periodicity detection and estimation.

Chapter 3

Motif Discovery in Nucleic Acid Sequences

3.1 Introduction

Gene expression underlies most essential cellular processes and is typically controlled by complex networks of regulatory interactions. Two of the basic mechanisms directly involved in regulating gene expression are transcription factor binding and site-specific recombination [33]. In both cases, the proteins involved often attach to highly specific nucleic acid sequences, which leads to the activation or repression of gene expression either through epigenetic interactions between transcription factors and components of RNA polymerase machinery or via recombinase-mediated genetic and genomic modifications of relevant DNA regions.

As individual binding sites are subject to context-specific optimizations of protein affinities as well as neutral alterations by random mutagenesis, nucleotide sequences of various site instances can display a significant de-

gree of heterogeneity. Even so, each instance may be expected to preserve certain core sequence features—such as nucleotide patterns responsible for the specificity of transcription factor binding or relative positions of bases where recombinase-induced DNA strand breaks can occur—making them identifiable as a motif. A key question in understanding the genomic organization and gene-regulatory network structure of biological systems thus comprises the discovery of conserved motifs within available sequence data. Still, although nucleic acid motif discovery (whereby one attempts to infer the identity and locations of conserved patterns in a given set of nucleotide sequences) has been the subject of much research in recent years, it remains a highly multifaceted and computationally challenging problem [34].

The principal subject of this chapter is further development of basic methodology for motif discovery within nucleic acid sequences. Following the discussion in Tompa et al. [34], we focus on analyzing primary sequence data—in the absence of any auxiliary information. Notably, this does not preclude but rather encourages the subsequent integration of our method with other heterogeneous approaches - such as those involving comparative sequence analysis, expression level data, chromatin immunoprecipitation results, and others - that synergistically complement each other by identifying interactions across different scales and domains of system organization. (For example, the cMonkey scheme successfully combines motif discovery by the antecedent MEME algorithm [35] with novel developments in biclustering

of expression data to generate cumulative improvements in gene regulatory network predictions [36].)

Along with performance, one of the essential requirements for a biologically-useful discovery algorithm is its broad applicability—both with respect to the lack of constraints on motif features as well as the universality of supported sequence databases. For instance, while a number of techniques have been developed for identifying a motif that appears only once in each sequence of a database, the same motif may and often has to be present at multiple sites in the genome. This is particularly significant in the case of recombinases, like those of the Din family, that require two or more separate sites to provide counterparts for strand exchange as well as in the case of primary regulon mediators, like cAMP-CRP, that must have multiple genomic targets in order to enable the sophisticated control patterns observed [33]—thus demanding that the motif discovery algorithm be able to identify several instances of the same motif in a given sequence. Furthermore, based on the extent of experimental evidence, the method should also accommodate scenarios where *a priori* knowledge of such motif features as length or composition is likely to either be incomplete, uncertain, or even entirely absent. The algorithm also needs to be versatile and scalable to be of meaningful practical utility. For example, since motif instances may be located near as well as far from any gene transcriptional start site, the technique must be capable of handling long sequences as well as short ones.

Many previously proposed solutions have been pattern-driven exhaustive searches, with the motif discovery question stated as an (l, d)-motif problem [37]. In this approach, the motif is assumed to be of length l and have at

most d mismatches between the true/empirical consensus sequence and its individual instances. Examples are WINNOWER [37], where the solution reduces to finding large cliques in multipartite graphs; and CONSENSUS [38], which uses a greedy technique to solve the problem. Another variant of this methodology is a sample-driven search that trades off sensitivity for computational efficiency by looking for patterns hidden in data subsets – such as employed by YMF [39], an enumerative algorithm that looks for motifs with highest z-scores; and Weeder [40], which uses extended enumeration that is better adapted to longer patterns. While potentially highly accurate, the main shortcoming of such methods is that they do not scale well with the size of the site, effectively limiting pattern-driven approaches to motifs no longer than 10 to 12 nucleotides [41].

An alternative is offered by profile-based methods that model motifs in statistical terms. A motif is then described by a position weight matrix (PWM), where each column relates to the distribution of all possible nucleotides at a given position. That is, in the case of DNA-drawn sequences and a motif of length M, the PWM is typically a $4 \times M$ matrix (often graphically represented as a logo), whose columns correspond to probability vectors of finding A, T, C, or G at the corresponding nucleotide position. This matrix is not known *a priori* and is usually estimated before or jointly with the discovery of locations of individual motif instances. Examples of such technique are MEME (Multiple EM for Motif Elicitation) [35, 42, 43], which utilizes expectation-maximization (EM) framework to discover an unknown number of different motifs that appear an unknown number of times; several algorithms—including BioProspector [44], AlignACE [45], Gibbs Motif

Sampler [46], MotifSampler [47], and SeSiMCMC [48]—that rely on Gibbs sampling; and Liang et al.'s approach [49], where a deterministic sequential Monte Carlo-based method is developed.

In this chapter, we present a Bayesian Algorithm for Multiple Biological Instances of motif discovery (BAMBI), which is able to detect an unknown motif of an unknown length with an unknown number of instances in a sequence database. The algorithm uses a profile-based approach—modeling a motif via PWM, which is estimated concurrently with the discovery task—and can work solely on the basis of nucleotide sequence data. (However, if additional experimental evidence, results of alternative motif discovery algorithms, or other sources of prior knowledge regarding any PWM components are available, BAMBI is flexible-enough to be able to include this information in its analysis.) Unlike earlier works, such as Liang et al. [49] that has developed a deterministic sequential Monte Carlo algorithm, our approach is able to independently estimate the putative motif size as well as to discover its multiple instances or to establish their absence in each of the database sequences – all within the Bayesian framework. The resulting method, BAMBI, displays better statistical performance than MEME, BioProspector (which is augmented with BioOptimizer [50] wherever there is uncertainty about motif length), SeSiMCMC, and Motif Sampler in three diverse settings, including being the only algorithm that leads to a biochemically meaningful result in the recombinase binding site discovery case.

3.2 Bayesian Algorithm for Multiple Biological Instances

With the Bayesian Algorithm for Multiple Biological Instances of motif discovery (BAMBI), we are seeking to discover nucleotide motifs, which are sets of patterns conserved when compared to a collection of nonspecific genomic segments. A database of nucleotide sequences—where each sequence may contain one, several, or no instances of motif—along with an upper limit on the total number of such instances in each sequence serve as problem inputs. For example, in the case of the CRP database (discussed later in further detail) the supplied input is a set of 105 nucleotide-long DNA segments from non-coding regions upstream of 18 *Escherichia coli* genes. The desired output is the number, length, and locations of CRP-binding sites within each sequence.

3.2.1 Overview

As noted earlier, the innate heterogeneity observed among instances of individual binding sites—which is driven by local context optimization requirements, mutagenesis, fluctuations in measurement fidelity, etc.—makes the determination of motif sequences a statistically uncertain problem. While these variations may be ascribed to an amalgamation of random processes, the ensuing probabilistic nature of the motif discovery problem can be captured through the use of the Hidden Markov Model (HMM) framework. That is, given a database of nucleic acid strand segments, we consider the information in question—namely, the number, length, and locations of individual motif instances in each sequence—to be unobservable directly (i.e.,

"hidden"). Instead, the available data consists solely of base sequences themselves, wherein motif patterns of interest – which remain to be "discovered" – may (or may not) be embedded. The approach used for the discovery process is based on Bayesian inference – a powerful and flexible technique able to utilize a broad range of data toward elucidating various hidden/unknown system parameters – which, in our case, focuses on motif lengths, logos, and instance locations. (Therein, one starts with a probabilistic model that reflects the knowledge regarding parameter values of interest as available *a priori*, if any. This 'prior' distribution is then updated to the 'posterior' one by conditioning on any additionally obtained information through the use of Bayes' probability formula, which results in *a posteriori* estimates of parameters that are progressively more constrained with each new observation.)

Significantly, although Bayesian techniques have been previously applied to the problem of identifying patterns in nucleic acid sequences, BAMBI implements this approach by treating entire sequences contained in the database (rather than single bases or smaller segments within them) as individual observations.

However, while generally more informative, the use of such larger data elements comes with substantial additional computational costs, which inhibit efficient model estimation. Here, we overcome this impediment through the use of a sequential Monte Carlo technique. This approach generates estimates of hidden variables by finding approximations of their posterior distribution given observations. Ideally, one might have liked to approximate this posterior distribution by obtaining samples from it, but this is generally

impossible—e.g., due to the referenced computational complexity. Instead, samples (called "particles") are first drawn from an alternative distribution (called "importance distribution") and a weight is then attached to each sample in such a way as to compensate for any mismatch between the true posterior and the importance distribution, which completes the method. (Given the broad freedom in choosing the importance distribution, here we have selected one that is suitable for a sequential method – that is, it enables processing of each observation individually.)

3.2.2 System Model and Problem Statement

We represent the system as a hidden Markov model (HMM) and process one sequence from the input database at a time in a sequential manner. The hidden state corresponds to the concatenation of the number of motifs in the current sequence and their initial locations, while the t-th sequence itself is considered to be the observation at time t. Moreover, given the state, the emission probability is considered to be dependent on an unknown position weight matrix (PWM), which describes the distribution of nucleotides in each position of the motif. A background distribution for the nucleotides not belonging to an instance of the motif is assumed to be given (e.g. by collecting sequence statistics of embedding DNA or by using results of other methods as input). In what follows, a mathematical description of the model and the problem statement are presented.

Let $\boldsymbol{S}_T = \{\boldsymbol{s}_1, \ldots, \boldsymbol{s}_T\}$ be the set of T sequences in the input database, used to learn the common motif, with $\boldsymbol{s}_t = \left[s_{t,1}, \ldots, s_{t,L(t)}\right]$ the t-th sequence of the database of length $L(t)$. Given an alphabet χ of size $|\chi|$, the

distribution of nucleotides in the M-long motif is considered unknown and it is described by a $|\chi| \times M$ PWM, $\boldsymbol{\theta} = [\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_M]$, which the algorithm has to estimate. Each $\boldsymbol{\theta}_j = \left[\theta_{j,1}, \ldots, \theta_{j,|\chi|}\right]$ with $j = 1, \ldots, M$ is the probability distribution of the letters in the alphabet for the jth position of the motif. If the sequences consist of DNA nucleotides, the alphabet is given by $\chi = \{A, C, G, T\}$ and $|\chi| = 4$. In this chapter, the nucleotides not belonging to a motif-region are assumed to be independent and identically distributed according to a background distribution given by $\boldsymbol{\theta}_0 = \left[\theta_{0,1}, \ldots, \theta_{0,|\chi|}\right]$. However, more complicate nucleotide models can be similarly used.

In addition, the number n_t of instances of a motif in each sequence is also taken as unknown and needs to be estimated. The distribution of the number of instances is described by the unknown vector $\boldsymbol{\lambda} = [\lambda_0 \ldots \lambda_N]$, where λ_j is the proportion of sequences with j instances of the motif and N is an upper bound on the number of instances.

At each step t we aim at estimating the state vector $\boldsymbol{x}_t$ composed by the number of motifs n_t present in the t-th sequence and the n_t initial positions of each instance of the motif in the sequence. Notice that the dimension of the vector $\boldsymbol{x}_t$ is not fixed and depends on n_t.

Given the sequences from first to t-th, $\boldsymbol{S}_t = \{\boldsymbol{s}_1, \ldots, \boldsymbol{s}_t\}$, and the distribution of nucleotides in the non-motif regions, we aim to discover the number of motifs in each sequence and their starting points, $\boldsymbol{X}_t = \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_t\}$. In this chapter, we propose to infer $\boldsymbol{X}_t$ within a Bayesian framework by modeling the position weight matrix $\boldsymbol{\theta}$ as Dirichlet random vectors, which provides additional information about base variations at each position in the motif across all of its instances within the database, and the distribution of the

number of instances $\boldsymbol{\lambda}$ of the motif in each sequence as a Dirichlet vector. The method is then further extended to solving the problem even if the length of the motif is unknown as well.

In the next section, we derive the sequential Monte Carlo method that will be used to solve the motif discovery problem.

3.2.3 Sequential Monte Carlo Method

Consider the general dynamic system with hidden state variable $\boldsymbol{x}_t$ and measurement variable $\boldsymbol{s}_t$, where there is an initial state model, i.e., $p(\boldsymbol{x}_0)$, and $\forall t \geq 1$, a state transition model, i.e., $p(\boldsymbol{x}_t|\boldsymbol{x}_{t-1})$, and a measurement model, i.e., $p(\boldsymbol{s}_t|\boldsymbol{x}_t)$. The sequence $\boldsymbol{X}_t = \{\boldsymbol{x}_1 ... \boldsymbol{x}_t\}$ is not observed and we want to estimate it for each time t, given that the measurements $\boldsymbol{S}_t = \{\boldsymbol{s}_1 ... \boldsymbol{s}_t\}$ are observed. If the distribution of the state from the initial time to time t given the observations, i.e., $p(\boldsymbol{X}_t|\boldsymbol{S}_t)$, is known, then many different estimators can be implemented. However, in the general case, computing $p(\boldsymbol{X}_t|\boldsymbol{S}_t)$ has a high complexity associated with it and approximations are used.

If samples from $p(\boldsymbol{X}_t|\boldsymbol{S}_t)$ were available, such distribution could be easily approximated, e.g., by using a Parzen window method [51]. However, getting samples from $p(\boldsymbol{X}_t|\boldsymbol{S}_t)$ is usually not feasible. An estimate can still be implemented by taking K samples $\boldsymbol{X}_t^k$ from a trial density $q(\boldsymbol{X}_t|\boldsymbol{S}_t)$. The support of the trial (or importance) distribution has to include the support

of $p(\boldsymbol{X}_t|\boldsymbol{S}_t)$. For the approximation, a weight is associated to each sample as follows,

$$w_t^k = \frac{p(\boldsymbol{X}_t^k|\boldsymbol{S}_t)}{q(\boldsymbol{X}_t^k|\boldsymbol{S}_t)}.$$

Each $\boldsymbol{X}_t^k$ $(k = 1, \ldots, K)$ is called particle or stream and the pair $\left\{\left(\boldsymbol{X}_t^k, w_{1:t}^k\right), k = 1, \ldots, K\right\}$ is said to be properly weighted with respect to the distribution $p(\boldsymbol{X}_t|\boldsymbol{S}_t)$. The approximation $\hat{p}(\boldsymbol{X}_t|\boldsymbol{S}_t)$ is then given by

$$\hat{p}(\boldsymbol{X}_t|\boldsymbol{S}_t) = \frac{1}{\sum_{j=1}^K w_t^j} \sum_{k=1}^K w_t^k \; \delta(\boldsymbol{X}_t - \boldsymbol{X}_t^k), \tag{3.1}$$

where $\delta(\boldsymbol{v})$ is 1 when $\boldsymbol{v} = \boldsymbol{0}$ and 0 everywhere else.

A sequential algorithm can be obtained by setting

$$q(\boldsymbol{X}_t^k|\boldsymbol{S}_t) = q(\boldsymbol{X}_{t-1}^k|\boldsymbol{S}_{t-1}) q(\boldsymbol{x}_t^k|\boldsymbol{X}_{t-1}^k, \boldsymbol{S}_t),$$

and noticing that the weights can be computed according to

$$w_t^k \propto w_{t-1}^k \frac{p(\boldsymbol{s}_t|\boldsymbol{X}_t^k, \boldsymbol{S}_{t-1}) p(\boldsymbol{x}_t^k|\boldsymbol{X}_{t-1}^k, \boldsymbol{S}_{t-1})}{q(\boldsymbol{x}_t^k|\boldsymbol{X}_{t-1}^k, \boldsymbol{S}_t)}. \tag{3.2}$$

Moreover, in order to minimize the variance of the weights, i.e., $var\left\{w_t|\boldsymbol{X}_{t-1}^k, \boldsymbol{S}_t\right\}$, the trial distribution can be chosen to be $q(\boldsymbol{x}_t^k|\boldsymbol{X}_{t-1}^k, \boldsymbol{S}_t) = p(\boldsymbol{x}_t^k|\boldsymbol{X}_{t-1}^k, \boldsymbol{S}_t)$, and the weights become

$$w_t^k \propto w_{t-1}^k p(\boldsymbol{s}_t|\boldsymbol{X}_{t-1}^k, \boldsymbol{S}_{t-1}). \tag{3.3}$$

The variance of the weights, however, increases over time which is known as the degeneracy phenomenon [51]. One option against this is to perform resampling to discard ineffective samples and multiply the effective ones.

When the state vector $\boldsymbol{x}_t$ can take a finite set of values, the sequential importance sampling (SIS) procedure with optimal importance distribution of [52] can be used. However, when the measurement model depends on an unknown vector $\boldsymbol{\theta}$, i.e., $p(\boldsymbol{s}_t|\boldsymbol{x}_t, \boldsymbol{X}_{t-1}^k, \boldsymbol{S}_{t-1}, \boldsymbol{\theta})$, and the state transition depends on a vector $\boldsymbol{\lambda}$, i.e., $p(\boldsymbol{x}_t|\boldsymbol{X}_{t-1}^k, \boldsymbol{S}_{t-1}, \boldsymbol{\lambda})$, it is possible to average out their influence. Therefore, the SIS procedure of [52] has to be modified to take this into account as follows.

For each time step, and for every particle, draw a sample $\boldsymbol{x}_t^k$ from $p(\boldsymbol{x}_t|\boldsymbol{X}_{t-1}^k, \boldsymbol{S}_t)$, where

$$\begin{aligned} p(\boldsymbol{x}_t|\boldsymbol{X}_{t-1}^k, \boldsymbol{S}_t) &\propto p(\boldsymbol{s}_t|\boldsymbol{x}_t, \boldsymbol{X}_{t-1}^k, \boldsymbol{S}_{t-1}) p(\boldsymbol{x}_t|\boldsymbol{X}_{t-1}^k, \boldsymbol{S}_{t-1}) \\ &= \int p(\boldsymbol{s}_t|\boldsymbol{x}_t, \boldsymbol{X}_{t-1}^k, \boldsymbol{S}_{t-1}, \boldsymbol{\theta}) p(\boldsymbol{\theta}|\boldsymbol{x}_t, \boldsymbol{X}_{t-1}^k, \boldsymbol{S}_{t-1}) d\boldsymbol{\theta} \\ &\qquad \int p(\boldsymbol{x}_t|\boldsymbol{X}_{t-1}^k, \boldsymbol{S}_{t-1}, \boldsymbol{\lambda}) p(\boldsymbol{\lambda}|\boldsymbol{X}_{t-1}^k, \boldsymbol{S}_{t-1}) d\boldsymbol{\lambda}, \end{aligned} \tag{3.4}$$

and let $\boldsymbol{X}_t^k = \left(\boldsymbol{X}_{t-1}^k, \boldsymbol{x}_t^k\right)$. Then update the importance weight as

$$w_t^k \propto w_{t-1}^k \sum_{\boldsymbol{x}_t} p(\boldsymbol{s}_t|\boldsymbol{x}_t, \boldsymbol{X}_{t-1}^k, \boldsymbol{S}_{t-1}) p(\boldsymbol{x}_t|\boldsymbol{X}_{t-1}^k, \boldsymbol{S}_{t-1}). \tag{3.5}$$

Finally, resample if needed.

In the next section we show how this set of expressions can be computed in closed forms for the problem of finding instances of a motif in a set of unaligned sequences.

3.2.4 Multiple Instance Motif Discovery Algorithm in a Bayesian Framework

Given the general solution of the sequential Monte Carlo method, in this section we particularize the solution to the specific case of an unknown number of instances of a motif of length M present in a database. We then extend the solution to the case where the length of the motif is not known.

Let $\boldsymbol{a}_{t,i}$ be the subsequence of M letters of the sequence $\boldsymbol{s}_t$ starting at position i, and $\boldsymbol{a}^c_{t,\boldsymbol{x}_t}$ the sequence resulting from removing $\boldsymbol{a}_{t,i}$ for $i = 1, \ldots, n$ from $\boldsymbol{s}_t$. Then, given the PWM $\boldsymbol{\theta}$, the background distribution $\boldsymbol{\theta}_0$ and the state at time t, the likelihood of a sequence $\boldsymbol{s}_t$ when $\boldsymbol{x}_t = [n, i_1, \ldots, i_n]$ is given is

$$p(\boldsymbol{s}_t|\boldsymbol{x}_t, \boldsymbol{X}_{t-1}, \boldsymbol{S}_{t-1}, \boldsymbol{\theta}) = \boldsymbol{\theta}_0^{\boldsymbol{n}(a^c_{t,\boldsymbol{x}_t})} \prod_{m=1}^{M} \boldsymbol{\theta}_m^{\sum_{j=1}^{n} \boldsymbol{n}(a_{t,i_j}(m))}, \tag{3.6}$$

where $\boldsymbol{\theta}^{\boldsymbol{n}} = \prod_{j=1}^{|\chi|} \theta_j^{n_j}$, $\boldsymbol{n}(\boldsymbol{a}) = \left[n_1, \ldots, n_{|\chi|}\right]$ with n_j for $j = 1, \ldots, |\chi|$ the number of times the jth letter appears in the sequence $\boldsymbol{a}$, and $\boldsymbol{a}(m)$ is the m-th letter of sequence $\boldsymbol{a}$.

For each position of the motif, an independent Dirichlet distribution [53] is used. This distribution has a well studied covariance structure and admits closed form expressions for its moments. The Dirichlet distribution has previously been used for modeling the PWM [49].

It is possible to use a sufficient statistic to update the distribution of the parameter $\boldsymbol{\theta}$ and to do it sequentially. Let $p(\boldsymbol{\theta}|\boldsymbol{X}_{t-1}, \boldsymbol{S}_{t-1})$ be the product of M independent random vectors distributed according to a Dirichlet

distribution, where the mth vector corresponds to the mth position in the motif and having parameters $\boldsymbol{\alpha}_m^{t-1} = \left[\alpha_{m,1}^{t-1} \dots \alpha_{m,|\chi|}^{t-1}\right]^T$, then $p(\boldsymbol{\theta}|\boldsymbol{X}_t, \boldsymbol{S}_t)$ is also the product of M Dirichlet distributions where the m-th distribution has, for $l = 1, \dots, |\chi|$, parameters

$$\alpha_{m,l}^t = \alpha_{m,l}^{t-1} + \sum_{j=1}^{n} n_l(a_{t,i_j}(m)). \tag{3.7}$$

Therefore, only the parameters of the M Dirichlet distributions need to be saved and are easily updated from time $t-1$ to time t. We define $\boldsymbol{T}_t^{\boldsymbol{\theta}}$ as a sufficient statistic to characterize the distribution $p(\boldsymbol{\theta}|\boldsymbol{X}_t, \boldsymbol{S}_t)$, that in our case is given by

$$\boldsymbol{T}_t^{\boldsymbol{\theta}} = \left\{\alpha_{m,l}^t\right\}_{m=1\dots M}^{l=1\dots|\chi|}, \tag{3.8}$$

which is a simple function of $\boldsymbol{T}_{t-1}^{\boldsymbol{\theta}}$ as shown in (3.7).

A sufficient statistic for the distribution of the number of instances $\boldsymbol{\lambda}$ of the motif in each sequence is also found. For this unknown vector, given a Dirichlet distribution with parameter $\boldsymbol{\gamma}_{t-1}$ for $p(\boldsymbol{\lambda}|\boldsymbol{X}_{t-1}, \boldsymbol{S}_{t-1})$, the distribution $p(\boldsymbol{\lambda}|\boldsymbol{X}_t, \boldsymbol{S}_t)$ is a Dirichlet distribution with parameter $\boldsymbol{\gamma}_t = \left[\gamma_0^t \dots \gamma_N^t\right]$, with $\boldsymbol{\gamma}_t = \boldsymbol{\gamma}_{t-1} + \boldsymbol{j}(\boldsymbol{x}_t)$, where $\boldsymbol{j}(\boldsymbol{x}_t)$ is a vector of zeros except for a 1 indicating the number of instances of the motif in the t-th sequence. We then have $\boldsymbol{T}_t^{\boldsymbol{\lambda}} = \boldsymbol{\gamma}^t$ as a sufficient statistic, where it is seen that $\boldsymbol{T}_t^{\boldsymbol{\lambda}}$ is a function of $\boldsymbol{T}_{t-1}^{\boldsymbol{\lambda}}$.

Importance Distribution $p(\boldsymbol{x}_t|\boldsymbol{X}_{t-1}^k, \boldsymbol{S}_t)$

This subsection develops closed form expressions for the importance distribution given in (3.4). The first integral on the right-hand side can be computed using (3.6), noticing that

$$\begin{aligned}
&\int p(\boldsymbol{s}_t|\boldsymbol{x}_t, \boldsymbol{X}_{t-1}^k, \boldsymbol{S}_{t-1}, \boldsymbol{\theta})\; p(\boldsymbol{\theta}|\boldsymbol{x}_t, \boldsymbol{X}_{t-1}^k, \boldsymbol{S}_{t-1})\; d\boldsymbol{\theta} \\
&\qquad = \boldsymbol{\theta}_0^{\boldsymbol{n}(a_{t,\boldsymbol{x}_t}^c)} \int \prod_{m=1}^{M} \boldsymbol{\theta}_m^{\sum_{r=1}^n \boldsymbol{n}(a_{t,i_r}(m))} p(\boldsymbol{\theta}|\boldsymbol{x}_t, \boldsymbol{X}_{t-1}^k, \boldsymbol{S}_{t-1}) d\boldsymbol{\theta} \\
&\qquad = \boldsymbol{\theta}_0^{\boldsymbol{n}(a_{t,\boldsymbol{x}_t}^c)} \prod_{m=1}^{M} E\left[\boldsymbol{\theta}_m^{\sum_{r=1}^n \boldsymbol{n}(a_{t,i_r}(m))}\right], \qquad (3.9)
\end{aligned}$$

where the expectation is taken over a Dirichlet distributed random variable.

To compute (3.9), we need the following theorem.

Theorem 8 *The general moment function of a Dirichlet distribution with parameters* $\boldsymbol{\alpha} = [\alpha_1, \ldots, \alpha_k]$ *is given by*

$$E\left[\theta_1^{r_1} \ldots \theta_k^{r_k}\right] = \frac{B(\boldsymbol{\alpha} + \boldsymbol{r})}{B(\boldsymbol{\alpha})}$$

where $\boldsymbol{r} = [r_1 \ldots r_k]$ *and* $B(\alpha) = \frac{\prod_{i=1}^k \Gamma(\alpha_i)}{\Gamma(\sum_{i=1}^k \alpha_i)}$.

Proof:

$$\begin{aligned}
E\left[\theta_1^{r_1} \ldots \theta_k^{r_k}\right] &= \int \theta_1^{r_1} \ldots \theta_k^{r_k} \frac{1}{B(\boldsymbol{\alpha})} \prod_{i=1}^{k} \theta_i^{\alpha_i - 1}\; d\boldsymbol{\theta} \qquad (3.10) \\
&= \frac{B(\boldsymbol{\alpha} + \boldsymbol{r})}{B(\boldsymbol{\alpha})} \left(\int \frac{1}{B(\boldsymbol{\alpha} + \boldsymbol{r})} \prod_{i=1}^{k} \theta_i^{\alpha_i + r_i - 1}\; d\boldsymbol{\theta} \right),
\end{aligned}$$

where the term between parenthesis is the integration of a Dirichlet distri-

bution with parameters $\boldsymbol{\alpha} + \boldsymbol{r}$ and therefore, the integral is 1. ■

If the prior for the generalized Dirichlet distributions consist of positive integers, then the αs and βs remain positive integers, and the fact that $\Gamma(n) = (n-1)!$, for n positive integer, can be used to make the algorithm efficient.

The second integral of (3.5) can be computed analogously.

$$\begin{aligned} &\int p(\boldsymbol{x}_t|\boldsymbol{X}_{t-1}^k, \boldsymbol{S}_{t-1}, \boldsymbol{\lambda})\; p(\boldsymbol{\lambda}|\boldsymbol{X}_{t-1}^k, \boldsymbol{S}_{t-1})\; d\boldsymbol{\lambda} \\ &\qquad = \quad p(i_1 \ldots i_n|n)\; E\left[\lambda_n\right] = p(i_1 \ldots i_n|n)\; \frac{\gamma_n^{t-1}}{\sum_{i=0}^{N} \gamma_i^{t-1}}, \end{aligned} \tag{3.11}$$

where $p(i_1 \ldots i_n|n)$ is taken to be uniform.

Then, given a database of sequences, the **B**ayesian **A**lgorithm for **M**ultiple **B**iological **I**nstance motif discovery algorithm is summarized as follows.

For each sequence, and for each particle,

- construct the importance distribution by enumerating all possible sample extensions

$$\boldsymbol{X}_t^k(n, i_1, \ldots, i_n) = \left[\boldsymbol{X}_{t-1}^k \;\; [n \; i_1 \;\; \ldots \;\; i_n]^T\right],$$

  and computing (3.5,3.9,3.11);

- sample $\boldsymbol{x}_t^k$ from the importance distribution and set

$$\boldsymbol{X}_t^k = \left[\boldsymbol{X}_{t-1}^k \; \boldsymbol{x}_t^k\right];$$

- compute the weight of the particle using (3.5), where each term of the

summation has already been computed in the first step;

Finally, update the sufficient statistics $\boldsymbol{T}_t^{\boldsymbol{\theta}}$ and $\boldsymbol{T}_t^{\boldsymbol{\lambda}}$ as shown in (3.7), and resample if needed.

3.2.5 Unknown Motif Length

The approach used here follows the class-based resampling scheme presented in [54]. In order to estimate the motif length m jointly with the number and locations of motifs for each sequence, we consider an augmented state vector $\boldsymbol{z}_t = \left[\boldsymbol{x}_t^T,\ m\right]^T$. As the length of the motif is not expected to change from sequence to sequence, a static dynamics is used for m. Moreover, to avoid letting the algorithm keep only particles with only one potentially incorrect motif length, we make sure that the method always saves particles for each of the possible considered motif lengths.

Therefore, given an upper and lower bound for the motif length, let Λ be the set of possible motif lengths. The resampling scheme can be summarized as follows.

- Choose the number of particles for each class N_m according to a multinomial distribution with parameters $\left\{\hat{P}(m|\boldsymbol{S}_t)\right\}_{m\in\Lambda}$.
- If the number of particles N_m is smaller than the threshold N_{thr}, set $N_m = N_{thr}$, and decrease the number of particles of the classes with most particles until $\sum_{m\in\Lambda} N_m = K$.
- Sample N_m new particles from the set of previous particles of the class with probabilities proportional to their weights. Assign equal weights to this particles within a class, i.e., $w_t^k = \hat{P}(m|\boldsymbol{S}_t)/N_m$.

3.2.6 Initializing Using Results From Another Motif Discovery Algorithm

The Bayesian framework proposed here is easily adaptable to use the results from another algorithm as a prior, and to refine the results of other motif discovery algorithms. If no other algorithm is available, the prior for the PWM is chosen to be an uninformative prior. However, if results from another motif discovery algorithm are available, the prior of the PWM can be easily modified to use this information. The estimated PWM by the other algorithm can be thought of as a Generalized Dirichlet distribution and with a sufficient statistic $\boldsymbol{T}_0^{\boldsymbol{\theta}}$.

The same can be done with $\boldsymbol{T}_0^{\boldsymbol{\lambda}}$ if there is prior knowledge of the distribution of the number of instances of the motif in the sequences.

3.2.7 Reduced Complexity Motif Discovery Alternative

As the number of particles needed to achieve a good performance increases with the dimension of the state vector, we propose to use the sequential Monte Carlo method to decide whether there is no instance of the motif or if there is only one instance of the motif in each sequence. This outputs an estimate of the PWM $\boldsymbol{\theta}$. To estimate the number of instances of the motif in each sequence, we propose to use the estimated $\boldsymbol{\theta}$ as a prior for a second stage where we use nested Neyman-Pearson (NP) hypothesis tests [15] as follows.

For each sequence, we compute a binary hypothesis test to determine whether the sequence has $j-1$ or j instances of the motif, starting with

$j = 1$. If we decide in favor of j, we increment j and retest. On the other hand, if we decide in favor of $j-1$, we then infer the locations of these instances by maximizing the likelihood of the observed sequence. For each binary hypothesis test, we use the NP test as it maximizes the probability of detection given an upper bound on the probability of false alarm. It proceeds as follows. Let $\mathcal{H}_{j-1}$ be the hypothesis that there are $j-1$ instances of the motif in the t-th sequence and $\mathcal{H}_j$ the hypothesis that the sequence has j instances of the motif. We can decide between the two hypotheses by computing the ratio

$$\frac{p(\boldsymbol{s}_t|\mathcal{H}_j)}{p(\boldsymbol{s}_t|\mathcal{H}_{j-1})} \underset{\mathcal{H}_{j-1}}{\overset{\mathcal{H}_j}{\gtrless}} \nu_j^t, \tag{3.12}$$

where ν_j^t is the threshold which is set to achieve a given probability of false alarm and can be found numerically.

When we decide in favor of the hypothesis $\mathcal{H}_{j-1}$, we estimate $\boldsymbol{x}_t$ by maximizing the likelihood given that there are $j-1$ instances of the motif. On the other hand, when we decide in favor of $\mathcal{H}_j$, we increment j and recompute the test in (3.12).

3.3 Experimental Results

We have applied BAMBI to several motif discovery problems, using both empirical as well as synthetic data, and evaluated its performance on the basis of the nucleotide-level correlation coefficient (nCC) — a robust mea-

sure that captures both the sensitivity and the specificity of a method [55]. While there are a number of alternative statistics that can potentially be used to compare performances of various bioinformatics algorithms, greatest nCC score has been suggested by Tompa et al. after an extensive study [34] as the reportable metric for subsequent assessment of motif discovery tools. It is defined as:

$$nCC = \frac{TP \cdot TN - FP \cdot FN}{\sqrt{(TP + FP)(FP + TN)(TN + FN)(FN + TP)}}$$

where TP/TN are the total number of nucleotides in the input database that are estimated to be true positives/negatives and FP/FN are the total number of nucleotides estimated to be false positives/negatives, based on an empirically established baseline standard.

In all instances, the performance of the presented algorithm has been further compared against four popular nucleic acid motif discovery methods: BioProspector, MEME, SeSiMCMC, and Motif Sampler.

In all the applications, BAMBI was initialized by setting the parameters of the corresponding Dirichlet distribution at each position in the PWM to be 1. This transforms the Dirichlet distribution into a uniform distribution, as no information about the motif is assumed. Similarly, the parameters of the Dirichlet distribution corresponding to the distribution of the number of instances of the motif in each sequence is initialized as follows. The parameter corresponding to the case of no instance of the motif is set to 1, and the parameter corresponding to the case of having one instance is set to be equal to the average length of the input sequences. This allows

the algorithm to have a good number of particles with an instance of the motif while having some with no instance as well when processing the first sequences. Finally, the number of particles is set to be 20 times the average length of the input sequences.

3.3.1 Synthetic database

Synthetic data was used to test each algorithm for different motif lengths. For every considered motif length, 10 databases were generated, each containing 25 sequences of 200 nucleotides. All sequences were seeded with 0, 1, or 2 instances of the motif with probabilities 0.1, 0.3 and 0.6 respectively. When a sequence has one or two instances of the motif, their locations are randomly selected using a uniform distribution. Nucleotides belonging to an instance of the motif were drawn from a distribution that has 0.7 probability for a dominant nucleotide and 0.1 for the remaining three nucleotides. The identity of the dominant nucleotide for each position was chosen randomly. For the positions in the sequence not belonging to a motif, the nucleotides are equiprobable, i.e., there is a probability of 0.25 for each nucleotide. The total nCC is computed for each motif lengths between 14 and 20.

The results produced by the BAMBI algorithm have been compared with those generated by MEME, BioProspector, SeSiMCMC, and Motif Sampler. All five algorithms have been given the exact motif length in each test. When applying Motif Sampler, the true background distribution is supplied as an input to the algorithm. The resulting values of nucleotide-level correlation coefficients are given as a function of motif length in Figure 3.1. It is seen that the algorithm proposed here achieves higher performance than the other

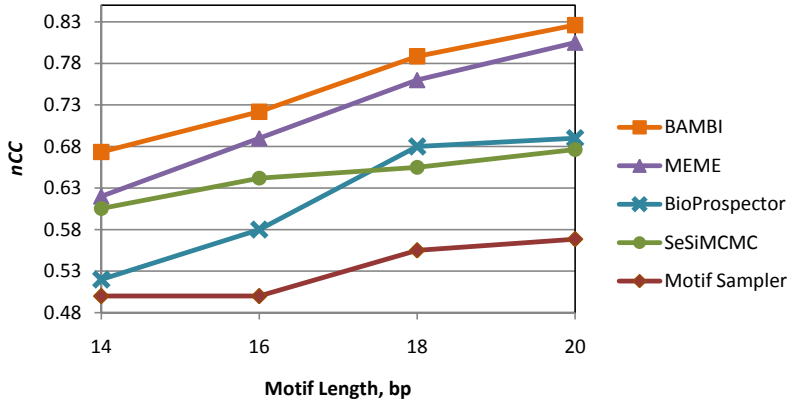


Figure 3.1: Performance comparison of different methods using synthetic data with varied motif length.

four methods for all tested motif lengths.

3.3.2 Real databases

We have analyzed two types of empirical DNA sequence data and compared the performance of BAMBI to that of MEME, BioProspector, SeSiMCMC, and Motif Sampler. The first application is a transcription factor binding site dataset, which consists of 18 short sequences that contain zero to two motif instances. The second is a site-specific recombinase binding dataset, which comprises only 10 sequences, but of considerably greater length each (see Table 3.2) that contain two instances of the motif each. This represents two completely different experimental scenarios where the Bayesian motif discovery is tested and compared with other approaches.

For these two datasets, we set Motif Sampler to estimate the background distributions as an order 1 Markov model from the input sequences. When analyzing the synthetic dataset, the true background distribution was supplied, but in the case of the real datasets, such distributions are unknown.

3.3.2.1 cAMP receptor protein (CRP) database.

Site-specific cAMP-CRP binding to DNA represents the prototypical model of gene regulation by a transcription factor [33, 56]. In large part, this may be attributed to CRP being an essential component of catabolite repression system, with research history in *E. coli* dating back to Monod's investigation of the "glucose effect" [56]. It also constitutes an example of a regulon, which plays a major role in directing bacterial energy metabolism [33] and whose significance has been recently further brought to fore by bioremediation and bioenergy applications [56, 57]. In fact, the identity of both CRP binding sites and amino-acid residues responsible for interacting with them have been so well-understood as to allow novel *in silico*-designed and *in situ*-engineered protein-DNA pairs binding with sufficient specificity to enable transcription factor activity [58]. Here, we apply BAMBI as well as MEME, BioProspector with BioOptimizer, SeSiMCMC, and Motif Sampler algorithms to identify the presence of CRP regulatory binding sites in 18 DNA sequences—each 105 nucleotides in length. It has been experimentally determined that there are 23 instances of the motif of length 22 in the set [59].

For the purposes of our analysis, the length m of the motif is considered to be unknown, requiring the use of respective procedures noted earlier. We impose a lower and upper bound on m of 17 and 27 – respectively – and set

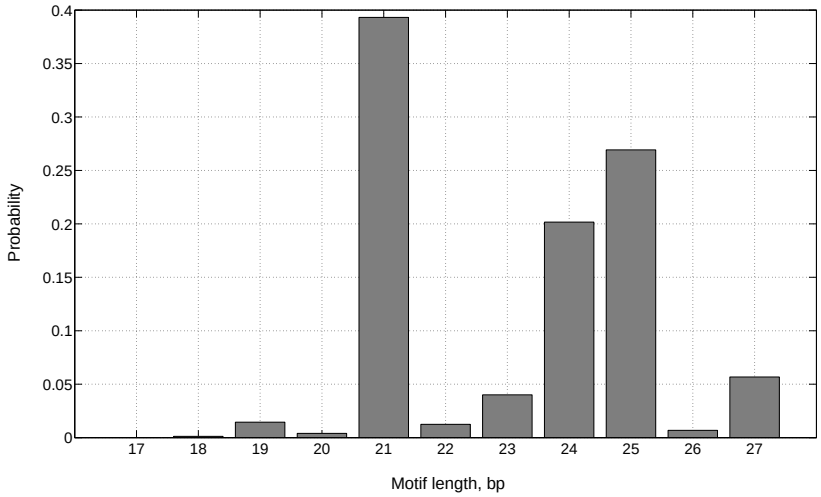
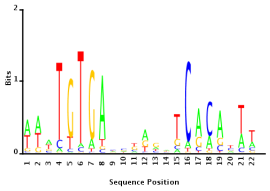


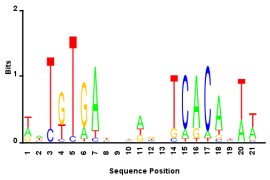
Figure 3.2: Motif Length PDF estimated by BAMBI for the CRP binding site motif.

the number of possible instances of the motifs to be between 0 and 2. (If another algorithm supplies more than two instances of a motif in a sequence, only the two highest scoring ones are kept to facilitate the comparison.) In the case of Motif Sampler, the length of the motif is supplied as an input to the method, as it cannot deal with uncertainty regarding this parameter.

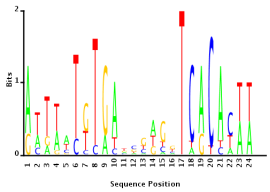
Figure 3.2 shows the estimated probability mass function of the different values of m after applying BAMBI to the entire database. As can be seen from the results, the BAMBI algorithm has estimated the most likely motif length to be 21bp-long, whereas the true motif length is considered to be 22bp, as noted earlier. By comparison, both MEME and BioProspector with BioOptimizer have estimated the length of the motif to be 24bp, with SeSiMCMC yielding 19bp.



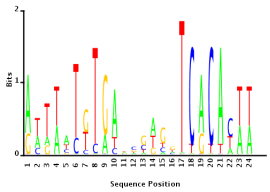
(a) True Motif Logo



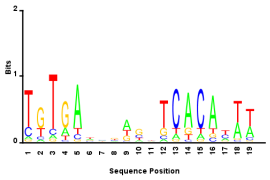
(b) BAMBI's Motif Logo



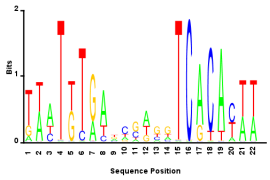
(c) MEME's Motif Logo



(d) BioProspector's Motif Logo



(e) SeSiMCMC's Motif Logo



(f) Motif Sampler's Motif Logo

Figure 3.3: Logos of the CRP binding site motif. Empirical ("True") versus those inferred by the different algorithms.

The estimated PWM logos for different motif discovery algorithms along with the one inferred from measured data are shown in Figure 3.3. The CRP motif contains two highly conserved inverted repeat sub-structures: "TGTGA" and "TCACA", which are likewise shown to be present in all of the logos.

The net results achieved by the BAMBI algorithm—as compared with those of MEME as well as BioProspector with BioOptimizer, SeSiMCMC, and Motif Sampler (with the latter having been supplied with known motif

length)—are given in Table 3.1, where $\hat{M}$ is the estimated motif length. It can be seen that BAMBI is performing better by both the statistical significance criterion (nCC) as well as based on the estimated motif length $\hat{M}$, for which BAMBI gives an estimate closest to the experimentally determined value.

Table 3.1: Performance comparison using the CRP database.

	BAMBI	MEME	BioProspector (+BioOptimizer)	SeSiMCMC	Motif Sampler
$\hat{M}$	21	24	24	19	-
nCC	0.6763	0.5358	0.5745	0.63633	0.5590

The value of M was found to be 22 empirically.

3.3.2.2 Din-family of site-specific serine recombinases database.

Site-specific recombination is a process by which well-defined sequences ("recombination sites") on the same or two different DNA molecules come together and undergo strand exchange, usually catalyzed by specialized enzymes called *recombinases* (sometimes contextually referred to as "invertases" or "integrases"). Based on the location/orientation of sites and other conditions, a recombination reaction results either in the inversion or excision/integration of the intervening DNA segment [60]. The latter generally contains promoters, alternative coding sequences, or other elements regulating gene expression; so that a recombination event causes initiation/cessation of transcription or/and synthesis of a different message RNA. Thus, site-specific recombination offers an organism or a virus an ability to generate mutually exclusive genetic states through "programmed"

DNA rearrangements. This type of gene regulatory mechanism has the advantage of being absolute—i.e., expression is impossible when the gene is lacking a correctly oriented promoter or is physically separated into several non-functional pieces—which may be critically important should presence of even one copy of the wrong protein become highly disadvantageous as, for example, might be the case for a pathogen targeted by antibodies directed against that protein [33, 61]. Recombination may also have a further advantage of facilitating rapid and optimized adaptation to such critical environmental conditions without the need to rely on slow and frequently deleterious process of random mutagenesis [62]. Indeed, gene regulatory networks driven by site-specific recombination appear to be particularly enriched among pathogens, including uropathogenic *Escherichia coli* – the predominant cause of urinary tract infections – and *Salmonella* Typhimurium [61, 62].

Importantly, such environmental conditions may often be rare or difficult to reproduce in the lab—e.g., when they involve intra-host pathogen dynamics [61]—causing potentially critical genomic rearrangements to remain phenomenologically undetected. One alternative could be to analyze genomic sequences directly for the presence of recombination sites through bioinformatics means. This approach may be further enabled by the fact that virtually all identified site-specific recombinases belong to one of just two basic families, named *serine* or *tyrosine* after the amino acid residue that forms the covalent protein-DNA linkage in the reaction intermediate [60]. The serine family comprises three primary subfamilies characterized by sequence, structural, and recombination site homology [62, 63]. Here, we

Table 3.2: Statistics of the recombinase database.

Number of Sequences	10
Shortest Sequence (nucleotides)	546
Longest Sequence (nucleotides)	4335
Average Sequence Length (nucleotides)	2436.4
Total Data set Size (nucleotides)	24364

use motif discovery algorithms to infer the DNA recombination site (*dix*) of Din serine subfamily, which includes such notable recombinase examples as Hin (responsible for flagellar phase variation in *Salmonella*), Gin (determination of phage Mu host specificity) as well as a number of other bacterial and phage systems.

All known Din family members recognize a 26bp-long minimal recombination sites [62, 64], with the list used in this study given in Table 3.3. Specific sequence sources employed to assemble the segment database used for site motif discovery comprised: *Salmonella enterica* serovar Typhimurium D23580 (GenBank FN424405); Bacteriophage Mu (GeneBank AF083977); Enterobacteria phage P1 (GenBank AF234172); prophage e14 of *Escherichia coli* K12 (GenBank K03521); *Escherichia coli* plasmid p15B (GenBank X62121); *Dichelobacter nodosus* VCS1001 (A198) (GenBank U02462); and *Shigella sonnei* (GenBank D00660 – revised from *S. boydii*, but functional in *S. sonnei* [65]). To generate the standardized data set, 7 sequences listed above were further cut, making sure two instances of the motif remained inside each segment. As there are 20 instances of the motif, this resulted in 10 sequences being used as the input to the algorithm. General characteristics of the so obtained database are shown in Table 3.4.

The number of nucleotides previous to the first instance of the motif is chosen from a uniform distribution between 0 and 50. The number of nucleotides to keep after the second instance of the motif was chosen analogously. Note that the two instances of the motif present in each sequence are often oriented in opposite directions, so the analysis has been extended in a straightforward manner to account for characteristics specific to double-stranded DNA by searching for sites located on the reverse complement as well. This is implemented within the context of the BAMBI hidden Markov model by replacing each double-stranded entry in the sequence database with one that is a concatenation of the corresponding forward and reverse strands (both in the $5'$-to-$3'$ orientation). As BAMBI is able to discover both the number and locations of multiple motif instances, running the algorithm over the modified database identifies sites located on either strand.

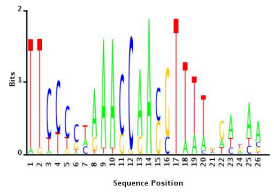
The logos estimated by the different algorithms are presented in Figure 3.4. It can be seen that BAMBI, MEME, and BioProspector find similar consensus sequences, while SeSiMCMC and Motif Sampler do not. A quantitative significance comparison of the results—given in Table 3.5—shows that the BAMBI algorithm achieves the best statistical performance, and that both SeSiMCMC and Motif Sampler were not able to find the motif.

Furthermore, only the BAMBI algorithm has been able to identify a functionally meaningful and biochemically correct recombination site. This is because, while for a transcription factor the inferred site only needs to specify preferred binding locations, in the recombinase case the DNA sequence itself has a functional role in gene expression regulation and so requires accurate identification of both the motif as well as strand breakage/exchange posi-

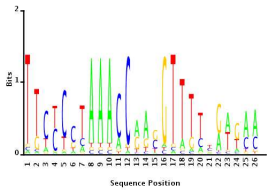
tions within it. As a result, any spatial shifts in the binding motif location away from the true sequence are likely to have a dramatic and deleterious effect on the product of site-specific recombination—e.g., by either putting an alternative coding sequence out of frame, removing a portion of the promoter region in the course of an inversion/excision or inhibiting strand exchange altogether. Thus, a shifted sequence prediction—no matter how close to the true motif in the statistical sense—cannot be deemed correct or acceptable in the biochemical sense as it undermines either bioengineering/synthetic biological implementation or systems biological analysis of the recombination products and their function.

In the case of the Din subfamily recombinase sites, the strand breakage/exchange reaction occurs through a staggered cut between the two "core" residues, which necessarily have to be symmetrically and centrally located within the recombinase binding motif (see Table 3.3 and, for example, [62]). As may be seen by comparing the inferred logos (Figure 3.4) among themselves or with the empirically established consensus Din binding site (Table 3.3), only the motif discovered by BAMBI accurately identifies the spatial location of the *dix* sequence, while the predictions of both MEME and BioOptimizer are shifted right by 3 bp. Given that the overall length of the motif is 26 bp, such a difference may not appear to be particularly significant statistically (e.g., as reflected by the nCC performance measure, Table 3.5). However, this is not the case biochemically, because such shifts generally lead to the incorrect determination of the identity of the two middle residues—the location of strand exchange—and so result in a non-functional recombinase site. For instance, outside of the two cen-

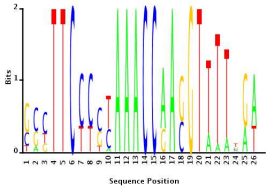
tral residues, the rest of the motif must largely be palindromic in order to accommodate the symmetric binding of two recombinase molecules, whose dimerization is generally required for strand exchange. However, in MEME- and BioOptimizer-discovered binding motifs, the lateral shift relative to the true empirically-known sequence substantially breaks this critical symmetry. Furthermore, the 2 bp central residue pair found via both MEME and BioOptimizer is a definitive **AC** (logo positions 13 and 14). However, the absence of complementary cores in the database as well as the presence of a "C" (instead of the strongly conserved "A", see Table 3.3) in the second position render such binding sites largely unable to support wild-type Din recombination, i.e., they are essentially non-functional [62]. These problems are notably not present in the BAMBI's motif prediction, which is spatially aligned with the *dix* sequence and assigns the most weight to either **AA** or **GA** core pairs that are biochemically permissible.



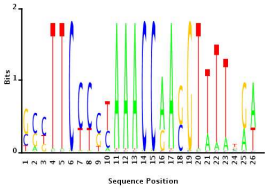
(a) True Motif Logo



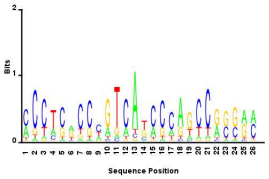
(b) BAMBI's Motif Logo



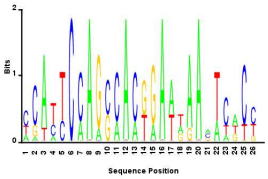
(c) MEME's Motif Logo



(d) BioProspector's Motif Logo



(e) SeSiMCMC's Motif Logo



(f) Motif Sampler's Motif Logo

Figure 3.4: Logos of the Din recombinase binding site motif. Empirical ("True") versus those inferred by the different algorithms.

Table 3.3: Target sites of Din-family recombinases.

dix (consensus)	TTC———AAAC–	**–A**	–GTTT———GAA
hixL	TTCTTGAAAACC	**AA**	GGTTTTTGATAA
hixR	TTTTCCTTTTGG	**AA**	GGTTTTTGATAA
gixL	TTCCTGTAAACC	**GA**	GGTTTTGGATAA
gixR	TTCCTGTAAACC	**GA**	GGTTTTGGATAA
cixL	TTCTCTTAAACC	**AA**	GGTTTAGGATTG
cixR	TTCTCTTAAACC	**AA**	GGTATTGGATAA
pixL	TTCTCCCAAACC	**AA**	GGTTTTCGAGAG
pixR	TTCTCCCAAACC	**AA**	CGTTTATGAAAA
mixMI"L'	TTCCCCCAAACC	**AA**	CGTTTTAGTCTT
mixMr"N'	TTCCCCTAAACC	**AA**	CGTTTTTATGCC
mixN"O'	TTCCCCCAAACC	**AA**	CGTTTTTATGTG
mixO"P'	TTCCCCTAAACC	**AA**	CGTTTTTATGCC
mixP"Q'	TTCCCCTAAACC	**AA**	CGTTTTTATGCC
mixQ"R'	TTCCCCCAAACC	**AA**	GGTAATCAAGAA
nix1	TTTCCCAGAAGC	**AA**	CCTTAAGTAAAA
nix2	TTTCGCAGAAGC	**AA**	CCTTACGTCAAA
nix3	AGACGAAGAAGC	**AA**	CCTTAAGTCAAA
nix4	TTTCCCAGAAGC	**AA**	CCTTAAGTCAAA
bixL	TTCCTGTAAACC	**GA**	GGTATTCGATAA
bixR	TTCCTGTAAACC	**GA**	GGTTTTAGATAA

Recombination sites for Din subfamily members: Hin (hixL and hixR), Gin (gixL and gixR), Cin (cixL and cixR), Pin (pixL and pixR), Min (mixMI"L', mixMr"N', mixN"O', mixO"P', mixP"Q' and mixQ"R' – labeled according to the convention used in [66]), *D. nodosus* (nix1, nix2, nix3 and nix4 – with sequences taken from the updated GenBank record rather than as specified in Moses et al. [64]), and PinB (bixL and bixR) [62, 64, 66–68]. Din palindromic consensus binding site (dix) is as discussed in [69]. The two core residues at the centers of the sites where strand breakage and exchange occur are highlighted in bold.

Table 3.4: Database of recombination sites.

GenBank Accession Number	Start Sequence	End Sequence	Recombination Sites
FN424405	2907699	2908805	hixL, hixR
AF083977	31913	35084	gixL, gixR
NC_005856	32206	36541	cixL, cixR
X01805	21	1929	pixL, pixR
X62121	2743	4447	mixR'M1", mixMr"N'
X62121	4848	5465	mixN"O', mixO"P'
X62121	5868	6414	mixP"Q', mixQ"L'
U02462	182	4049	nix1, nix2
U02462	4489	8411	nix3, nix4
D00660	600	3788	bixL, bixR

Sequence *start* and *end* labels are given by the nucleotide number in the corresponding GenBank record.

Table 3.5: Performance comparison using the recombinase database.

	BAMBI	MEME	BioProspector	SeSiMCMC	Motif Sampler
nCC	0.7711	0.7618	0.7618	-0.0153	-0.0182

MEME, BioProspector, SeSiMCMC and Motif Sampler did not produce a functionally correct site.

Chapter 4

Haplotype Inference

4.1 Introduction

Diploid organisms have two homologous copies of each chromosome, one inherited from the father and one from the mother. The two copies are not necessarily identical as there are loci in the genome where single nucleotides differ between members of the same species. These sites are called single nucleotide polymorphism (SNP). The SNPs are often located close to each other on the DNA and are inherited together as a set, and the sequence of nucleotides of that set in each of the two chromosome copies is called a haplotype.

The knowledge of each haplotype for an individual brings about improvements in drug design, diseases detection [70] and also provides useful information for evolutionary studies on populations [71]. However, direct measurement of the haplotypes is expensive and time consuming, and usually only the genotype is measured, i.e., a conflation of the haplotypes. For

each locus, the genotype contains information on the two occurring nucleotides, but it does not indicate in which of the two chromosome copies a particular nucleotide resides. Therefore, it is not possible to obtain the haplotypes of a person given its genotype. On the other hand, if the genotypes of a group of people are available, information from population genetics can be used to infer the haplotypes of each individual.

Approaches to solving the haplotype inference problem can be divided in two categories: rule-based methods and statistical methods. Rule-based methods rely on the maximum parsimony criterion [72] which states that the observed genotypes are generated by the minimum number of distinct haplotypes. This leads to a combinatorial problem that has been shown to be NP-hard [73, 74]. One approach to developing a fast method was presented in [75] where the so-called Clark's rule is applied iteratively by adding a single haplotype in each step to explain each unexplained genotype. The set of estimated haplotypes in this case depends on the order in which the genotypes are given. A generalization to Clark's rule was presented in [76], where another heuristic called CollHaps is used to find the haplotypes. However, Clarks rule does not yield an effective approximation algorithm for maximum parsimony [77]. RTIP [78] is a method that finds the maximum parsimony solution by solving an integer linear program whose size grows exponentially with the number of heterozygous positions in genotypes; HAPAR [79] is a similar technique that employs a branch-and-bound method initialized with the solution of a greedy algorithm; and in [80] the authors propose two methods based on a sequence of LP relaxations and the search for valid cuts.

On the other hand, there are several statistical methods for haplotype inference, among which PHASE [81] offers the best performance. It is a Bayesian algorithm that models the unknown haplotypes as unobserved random quantities. Given the observed genotypes, it approximates the conditional distributions by using a Gibbs sampler and the coalescence theory. However, it ends up with an approximation that is not a valid posterior [82]. The main drawback of this method is its slow speed. HAP [83] is a faster method that finds candidate solutions by using a perfect phylogeny procedure and picks the one achieving the greatest likelihood. A Gibbs sampler was also used to solve the haplotype inference problem in [84, 85]. In this case, a Dirichlet distribution is used as the prior for the vector of frequencies of the different haplotypes, allowing an excessive number of haplotype to be used which produces artifacts [82]. Haplotyper is another statistical method [86] based on the Gibbs sampler, but a prior annealing is used to ameliorate the problems of the Dirichlet prior. As an alternative to the Gibbs sampler, an expectation-maximization (EM) algorithm is used in [87] which is sensitive to the initial conditions, and is limited in the number of SNPs it can handle (on the order of 20). A partition-ligation EM method was later introduced to overcome this computational limitations [88]. Another example of haplotyping based on the EM algorithm is Gerbil [89] which identifies haplotypes and SNP blocks simultaneously. More recently, a new method called fastPHASE was presented in [90] where a clustering approach was used before the Gibbs sampler in order to obtain a faster algorithm than PHASE. In general, methods based on the Gibbs sampler and the EM algorithm are not robust when the parameter space shows multimodality, which

is the case for the haplotype inference problem [91]. In [92] the authors propose a method based on the deterministic sequential Monte Carlo approach to overcome this lack of robustness.

In this chapter, we first propose a new mathematical framework for haplotype inference based on the sparse representation of the observed genotypes. Within this framework, we present two related haplotype inference methods. In the first one, the maximum parsimony principle is translated to a sparseness condition on the haplotype frequency vector. We then propose to minimize the Tsallis entropy of this frequency vector in order to obtain a sparse solution. This leads to a method that relies on the minimization of a concave function which is also NP-hard. We present a method that enumerates all the local minima of the Tsallis entropy with high probability by solving a succession of integer linear programs. The solution is then found among the local minimum points with the smallest Tsallis entropy. The method contains a parameter that represents the tradeoff between accuracy and execution time. We then introduce a second method that looks for a dictionary of haplotypes to reconstruct all observed genotypes. The maximum parsimony principle is translated, in this case, to the search for a sparse dictionary. This leads to an approximately submodular optimization problem that can be solved efficiently with a simple greedy algorithm. We extend our method to handle long genotype vectors and missing data.

The remainder of the chapter is organized as follows. In Section 4.2, we introduce the novel mathematical framework for the haplotype inference problem. In Section 4.3, we represent the maximum parsimony principle as a sparseness condition on the haplotype frequency vector and introduce the

first method to perform the haplotype inference. Then, in Section 4.4, we present the second method that looks for a sparse dictionary and leads to a more efficient way of solving the haplotype inference problem. We provide experimental results on synthetic and real datasets in Section 4.6.

4.2 System Model and Problem Statement

A SNP is a single nucleotide variation where only two out of the four different nucleotides occur in a large percentage of the population. Then, only one of two states (alleles) can be found in a specific position of the chromosome (locus) of a SNP. The most common nucleotide in that locus is called the wild-type and is encoded with a 0 and the other nucleotide is the mutant and is encoded with a 1. When analyzing L SNPs, the states of the loci on each copy of the chromosome is represented separately as a haplotype, and therefore, diploid organisms have two haplotypes. If both haplotypes of an individual have a 0 (1) for a specific locus, the site is called homozygous and is encoded with a 0 (2). In this case, for this locus, we say that the genotype presents no ambiguity as the genotype can be used to reconstruct each haplotype unequivocally. On the other hand, when the alleles are different, the site is heterozygous and the genotype for the locus is 1. In this type of site, the genotype gives no information about which haplotype contains the wild-type and which one contains the mutant. We call this an ambiguity. Notice that this convention is slightly different from those in most previous works, but it allows us to express the genotype $g_i(\ell)$ of the i-th individual at the ℓ-th locus as the sum of the corresponding two

haplotypes $h_i^1(\ell)$ and $h_i^2(\ell)$, i.e.,

$$g_i(\ell) \quad = \quad h_i^1(\ell) + h_i^2(\ell), \quad \ell = 1, \ldots, L. \tag{4.1}$$

Let $\boldsymbol{h}_i^j = \left[h_i^j(1) \ldots h_i^j(L)\right]^T$ be the j-th haplotype of the i-th person, $j \in \{1, 2\}$, consisting of L SNP loci, where $h_i^j(\ell) \in \{0, 1\}$. Moreover, let $\boldsymbol{g}_i = [g_i(1) \ldots g_i(L)]^T$, with $g_i(\ell) \in \{0, 1, 2\}$, be the genotype data for that same individual. Then, for each individual $i = 1 \ldots N$, where N is the number of individuals in the input dataset, we have

$$\boldsymbol{g}_i \quad = \quad \boldsymbol{h}_i^1 + \boldsymbol{h}_i^2. \tag{4.2}$$

We say that a haplotype $\boldsymbol{z} \in \{0, 1\}^L$ is compatible with a genotype $\boldsymbol{g} \in \{0, 1, 2\}^L$ if $\boldsymbol{g} - \boldsymbol{z} \in \{0, 1\}^L$. We are interested in the haplotypes compatible with the genotypes as they are candidates to be selected as the inferred haplotypes for a given genotype. Given a set of genotypes, we define the haplotype dictionary matrix $\boldsymbol{Z}$, which has the haplotypes that are compatible with the observed genotypes as its columns. To obtain $\boldsymbol{Z}$ from the observed genotypes, we proceed as follows. For each observed genotype $\boldsymbol{g}_i$, we generate the set $\mathcal{H}_i$ of haplotypes that are compatible with $\boldsymbol{g}_i$. Then, $\boldsymbol{Z}$ has each haplotype of the union of the sets $\mathcal{H}_1, \ldots, \mathcal{H}_N$ in its columns. Let M be the number of haplotypes in $\boldsymbol{Z}$.

As $\boldsymbol{g}_i = \boldsymbol{h}_i^1 + \boldsymbol{h}_i^2$, both $\boldsymbol{h}_i^1$ and $\boldsymbol{h}_i^2$ are compatible with $\boldsymbol{g}_i$ and therefore, they belong to $\mathcal{H}_i$ and are columns of the matrix $\boldsymbol{Z}$. Let $\boldsymbol{h}_i^1$ and $\boldsymbol{h}_i^2$ be the r-th and s-th columns of the $L \times M$ matrix $\boldsymbol{Z}$ respectively. Then, we have

that genotype $\boldsymbol{g}_i$ of the i-th individual can be expressed as

$$\boldsymbol{g}_i = \boldsymbol{Z}\boldsymbol{x}_i, \tag{4.3}$$

where $\boldsymbol{x}_i \in \{0,1,2\}^M$ is a *sparse* vector that indicates the haplotypes generating the genotype $\boldsymbol{g}_i$ from the available dictionary of M haplotypes. More specifically, if $r \neq s$, then $x_i(j) = 0$ for $j = \{1 \dots M\} \backslash \{r, s\}$ and $x_i(r) = x_i(s) = 1$. Otherwise, if $r = s$, then $x_i(j) = 0$ for $j = \{1 \dots M\} \backslash \{r\}$ and $x_i(r) = 2$. Notice that $\mathbf{1}^T \boldsymbol{x}_i = 2$, where $\mathbf{1} = [1 \dots 1]^T$. Furthermore, we define the following haplotype frequency vector $\boldsymbol{f}(\boldsymbol{x}_1, \dots, \boldsymbol{x}_N)$.

$$\boldsymbol{f}(\boldsymbol{x}_1, \dots, \boldsymbol{x}_N) \triangleq \frac{1}{2N} \sum_{n=1}^{N} \boldsymbol{x}_n. \tag{4.4}$$

The maximum parsimony principle states that the number of different haplotypes that explains all the observed genotypes should be as small as possible. Therefore, the maximum parsimony haplotype inference problem is stated as follows. Given the set $\{\boldsymbol{g}_i, i = 1, \dots, N\}$ of genotype vectors of N subjects for L loci, we aim at inferring the set of haplotypes pairs $\{\boldsymbol{h}_i^1, \boldsymbol{h}_i^2, i = 1, \dots, N\}$ that is composed of the minimum number of distinct haplotypes, without the prior knowledge about the frequencies of the haplotypes. More specifically, we will infer $\boldsymbol{x}_i$, which given the matrix $\boldsymbol{Z}$, is equivalent to inferring the haplotypes $\{\boldsymbol{h}_i^1, \boldsymbol{h}_i^2\}$ of the i-th individual.

Our approach employs the maximum parsimony principle within the presented mathematical framework. This principle states that the solution should have a frequency vector with as few non-zero components as possi-

ble. Equivalently, from the point of view of the decomposition in (4.3), the principle requires that we need to use as few columns of $\boldsymbol{Z}$ as possible to explain all the observed genotypes.

4.3 Sparse Haplotyping based on Tsallis Entropy Minimization

4.3.1 Problem Formulation

Notice that the indicator vector for the genotypes that have no ambiguity can be found by searching through the columns of matrix $\boldsymbol{Z}$. Let $\boldsymbol{g}_i$ be a genotype with no ambiguities, then the i-th person has two identical haplotypes, i.e., $\boldsymbol{h}_i^1 = \boldsymbol{h}_i^2 = \boldsymbol{g}_i/2$. And the indicator vector $\boldsymbol{x}_i$ is determined by finding the column of matrix $\boldsymbol{Z}$ that equals to $\boldsymbol{g}_i/2$; that is, if the r-th column of $\boldsymbol{Z}$ is $\boldsymbol{g}_i/2$, then $x_i(j) = 0$ for $j = \{1 \dots M\} \backslash \{r\}$ and $x_i(r) = 2$. Moreover, if the i-th genotype has only one ambiguity, the corresponding haplotype pair can be easily found by setting $h_i^1(\ell) = 0$ and $h_i^2(\ell) = 1$ where the genotype presents an ambiguity, i.e., $g_i(\ell) = 1$, and $h_i^1(\ell) = h_i^2(\ell) = g_i(\ell)/2$ otherwise. Then the two different haplotypes $\boldsymbol{h}_i^1$ and $\boldsymbol{h}_i^2$ need to be found among the columns of $\boldsymbol{Z}$.

Therefore, we only need to infer the haplotype pair corresponding to genotypes with at least two ambiguities. Let $\mathcal{I}$ be the set of indices of genotypes with two ambiguities or more. Each individual $i \in \mathcal{I}$ has two different haplotypes, and therefore, $\boldsymbol{x}_i \in \{0, 1\}^M$. We propose to find the set of indicator vectors $\{\boldsymbol{x}_i, i \in \mathcal{I}\}$ by solving the following optimization

problem:

$$\max_{\boldsymbol{x}_i,\, i\in\mathcal{I}} \quad \text{parsimony} \qquad (4.5)$$

$$\text{subject to} \quad \begin{cases} \boldsymbol{g}_i = \boldsymbol{Z}\boldsymbol{x}_i \\ 2 = \mathbf{1}^T\boldsymbol{x}_i \\ \boldsymbol{x}_i \in \{0,1\}^M, \quad i \in \mathcal{I}. \end{cases}$$

For future reference, we denote the constraint set

$$\mathcal{S}_i \triangleq \left\{\boldsymbol{x}_i \in \{0,1\}^M : \boldsymbol{g}_i = \boldsymbol{Z}\boldsymbol{x}_i, 2 = \mathbf{1}^T\boldsymbol{x}_i\right\},$$

and

$$\mathcal{S} = \left\{\{\boldsymbol{x}_i\}_{i\in\mathcal{I}} : \boldsymbol{x}_i \in \mathcal{S}_i,\ i \in \mathcal{I}\right\}.$$

We translate the parsimony principle to a sparseness condition over the frequency vector $\boldsymbol{f}(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N)$ defined in (4.4). Such a condition can be expressed mathematically by means of the l_0 norm, which counts the number of non-zero elements of its argument. Therefore, minimizing $\|\boldsymbol{f}(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N)\|_0$ leads to a maximum parsimony solution.

However, minimizing the l_0 norm has an exponential complexity. A relaxation to the objective function can be applied in order to obtain a more tractable problem. In the compressed sensing literature, the l_1 norm is usually used as a substitute for the l_0 norm. But the l_1 norm of a frequency vector is always equal to one and therefore, it cannot be used here. We propose to use the Tsallis entropy [93] H_q with small $q > 0$ to induce the sparse condition. Let $\mathcal{F} \triangleq \{\boldsymbol{y} = [y_1 \ldots y_M]^T : \sum_{i=1}^M y_i = 1, y_i \geq 0\}$ be the

set of frequency vectors. The Tsallis entropy of a frequency vector $\boldsymbol{y} \in \mathcal{F}$ is defined as

$$H_q(\boldsymbol{y}) = \frac{1}{q-1}\left(1 - \sum_{i=1}^{M} y_i^q\right). \tag{4.6}$$

This entropy has the characteristic of being strictly concave for $q > 0$ and moreover, as it is symmetric, it is Schur-concave [94], i.e., if $\boldsymbol{x} \prec \boldsymbol{y}$[1], then $H_q(\boldsymbol{x}) \geq H_q(\boldsymbol{y})$. As a consequence of this property, the frequency vector that minimizes the Tsallis entropy can be found and consists of the sparsest possible vector. To see this, let $\boldsymbol{y}^* = [y_1^* \dots y_M^*]^T \in \mathcal{F}$ be defined as

$$y_i^* = \begin{cases} 1 & \text{if } i = k, \\ 0 & \text{otherwise,} \end{cases} \tag{4.8}$$

for any $k \in \{1 \dots M\}$. This vector has the property that for all $\boldsymbol{y} \in \mathcal{F}$, $\boldsymbol{y}^* \succeq \boldsymbol{y}$ and therefore, $H_q(\boldsymbol{y}^*) \leq H_q(\boldsymbol{y})$.

Hence the Tsallis entropy can be a good alternative to the l_0 norm. Moreover, it is seen from the definition of the entropy that if we make the parameter $q = 0$, the entropy disregards the values of the components of the vector $\boldsymbol{y}$ and only counts the number of nonzero components [95], i.e., $H_0(\boldsymbol{y}) = -1 + \|\boldsymbol{y}\|_0$. Therefore, we propose to minimize $H_q(\boldsymbol{y})$ with a small $q > 0$ as a concave approximation to minimizing $\|\boldsymbol{y}\|_0$. In Fig. 4.1 we show

[1]For any $\boldsymbol{x} = [x_1, \dots, x_M]^T \in \mathbb{R}^M$, let $x_{[1]} \geq \dots \geq x_{[M]}$ denote the components of $\boldsymbol{x}$ in decreasing order. Then, for $\boldsymbol{x}$, $\boldsymbol{y} \in \mathbb{R}^M$, $\boldsymbol{x}$ is majorized by $\boldsymbol{y}$, i.e., $\boldsymbol{x} \prec \boldsymbol{y}$

$$\text{if} \quad \begin{cases} \sum_{i=1}^{k} x_{[i]} & \leq & \sum_{i=1}^{k} y_{[i]}, & k = 1, \dots, M-1, \\ \sum_{i=1}^{M} x_{[i]} & = & \sum_{i=1}^{M} y_{[i]}. \end{cases} \tag{4.7}$$

$H_q(\boldsymbol{y})$ with $\boldsymbol{y} = [y_1\ y_2\ y_3]^T$ for four different values of q. As $y_3 = 1 - y_1 - y_2$, we display the entropy as a function of only y_1 and y_2. It is seen in the figure that the entropy is minimum when only one component of $\boldsymbol{y}$ is one and the remaining ones are zero, corresponding to the sparsest frequency vectors. It is also seen in the figure that as q goes to zero, the entropy goes to $\|\boldsymbol{y}\|_0 - 1$.

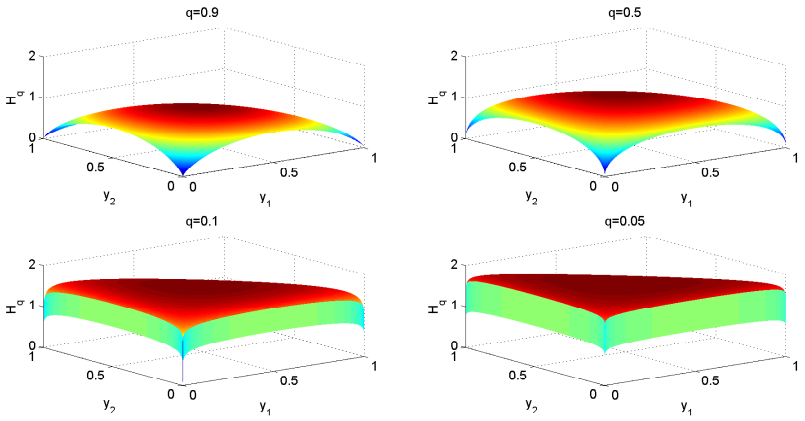


Figure 4.1: The Tsallis entropy for a frequency vector $\boldsymbol{y} = [y_1\ y_2\ y_3]^T$ for different values of q.

Therefore, minimizing the Tsallis entropy results in sparseness. Then the sparse haplotyping method based on the Tsallis entropy minimization is formulated as follows:

$$\begin{aligned} \min_{\boldsymbol{x}_i,\, i\in\mathcal{I}} \quad & H_q\left(\boldsymbol{f}(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N)\right) \\ \text{subject to} \quad & \boldsymbol{x}_i \in \mathcal{S}_i,\ i \in \mathcal{I}. \end{aligned} \tag{4.9}$$

4.3.2 Solution

The sparse haplotyping based on the Tsallis entropy minimization is an integer programming problem with a concave objective function. Notice that linear functions are a special case of concave functions, and integer linear programming problems are known to be NP-hard [96]. Therefore, our Tsallis entropy minimization problem is also NP-hard. Moreover, the minimization of a concave function may have many local solutions, and local optimality does not imply global optimality. Consequently, finding the global minimum is a computationally difficult problem. We present in this subsection a method to uncover the local minima in order to find the optimal solution among these points with high probability based on a multistart stochastic method. This method is based on an algorithm in [97] for solving linearly constrained concave global minimization problems.

Notice that (4.9) can be converted to a linearly constrained concave minimization problem as follows. First, S can be replaced with its convex hull $conv(S)$ as it is known that the global minimum point of a concave function over a nonempty and bounded convex polytope is always found at a vertex of the polytope [97] and the vertices of $conv(S)$ belong to S [98]. This allows us to rewrite the combinatorial problem as the following linearly constrained concave global minimization problem:

$$\begin{aligned} \min_{\boldsymbol{x}_i\ i\in\mathcal{I}} \quad & H_q\left(\boldsymbol{f}(\boldsymbol{x}_1,\ldots,\boldsymbol{x}_N)\right) \\ \text{subject to} \quad & \{\boldsymbol{x}_i,\ i\in\mathcal{I}\}\in conv(S), \end{aligned} \tag{4.10}$$

where the set of feasible points is non-empty and a bounded polytope. How-

ever, the set $conv(S)$ is not easily characterized in a closed-form expression but finding a solution to the optimization is possible nonetheless.

As the global minimum point of a concave function is always found at a vertex of the convex polytope, we use a stochastic multistart technique to list all vertices corresponding to local minima of the objective function. Then, finding the global minimizer is just a matter of looking through the set of local minima and identifying the one with the minimum objective function value. The method iterates between two phases. In the first or global phase, the search is done in a random direction to find a vertex of the polytope. Then, the second or local phase involves finding a local minimum starting from the solution to the global phase. These two phases iterate until all local minima are visited with high probability.

4.3.2.1 Global Search

The global search is carried out as a means of finding an initial point for the local phase. This is done by replacing the concave objective function by a linear function, with a random direction $\boldsymbol{u} \in \mathbb{R}^{MN}$, as follows,

$$\begin{aligned} \min_{\boldsymbol{x}_i\ i\in\mathcal{I}} \quad & \boldsymbol{u}^T \begin{bmatrix} \boldsymbol{x}_1 \\ \vdots \\ \boldsymbol{x}_N \end{bmatrix} \\ \text{subject to} \quad & \{\boldsymbol{x}_i,\ i \in \mathcal{I}\} \in conv(S). \end{aligned} \tag{4.11}$$

This problem is a linear program (LP), and its solution is a vertex of the polytope of feasible solutions.

However, we do not have a closed-form representation for $conv(S)$. Nonetheless, following the similar argument as before, we can replace $conv(S)$ with S and solve the equivalent linear integer optimization problem. Moreover, let $\boldsymbol{u} = \left[\boldsymbol{u}_1^T \dots \boldsymbol{u}_N^T\right]^T$ be partitioned into N vectors of dimension M, then (4.11) is equivalent to solving the following $|\mathcal{I}|$ smaller optimization problems,

$$\min_{\boldsymbol{x}_i} \quad \boldsymbol{u}_i^T \boldsymbol{x}_i \tag{4.12}$$

$$\text{subject to} \quad \begin{cases} \boldsymbol{g}_i = \boldsymbol{Z}\boldsymbol{x}_i \\ 2 = \mathbf{1}^T \boldsymbol{x}_i \\ \boldsymbol{x}_i \in \{0,1\}^M, \quad i \in \mathcal{I}, \end{cases}$$

which is a binary integer linear program that can be efficiently solved with a succession of linear programs under the framework of the branch-and-bound method [96].

Notice that this step is choosing a haplotype pair for each individual randomly. Then, the binary integer linear program can be replace by uniformly choosing a haplotype pair that explains the corresponding genotype of the i-th individual. This can be efficiently implemented by computing a list of possible haplotypes pair only once, and then uniformly choosing from this list. The complexity of the binary linear program is then avoided.

Let $\boldsymbol{z}_o = [\hat{\boldsymbol{x}}_i,\ i \in \mathcal{I}]$ be the stacked solution to the $|\mathcal{I}|$ random selection problems.

4.3.2.2 Local Search

The local phase uses the point $\boldsymbol{z}_0$ found by the global phase to find a local minimum of (4.10). This phase consists of a sequence of linear programs where given the solution in step $j-1$, the solution in step j is found by solving the following optimization problem.

Given a vertex $\boldsymbol{z}_{j-1}$, find another vertex with a smaller objective function value by solving the following linear program:

$$\begin{aligned} \min_{\boldsymbol{x}_i,\, i\in\mathcal{I}} \quad & \nabla\big(H_q\left(\boldsymbol{f}\left(\boldsymbol{z}_{j-1}\right)\right)\big)^T\big(\left[\boldsymbol{x}_i,\ i\in\mathcal{I}\right]-\boldsymbol{z}_{j-1}\big) \\ \text{subject to} \quad & \{\boldsymbol{x}_i,\ i\in\mathcal{I}\}\in conv(S), \end{aligned} \tag{4.13}$$

where $[\boldsymbol{x}_i,\ i\in\mathcal{I}]$ is the concatenation of the vectors $\boldsymbol{x}_i$ with $i\in\mathcal{I}$. Let the solution to this linear program be $\boldsymbol{z}_j$. It is shown in [97] that the point that solves (4.13) attains a lower objective function value, i.e., $H_q\left(\boldsymbol{f}\left(\boldsymbol{z}_j\right)\right)\leq H_q\left(\boldsymbol{f}\left(\boldsymbol{z}_{j-1}\right)\right)$. Iterate this step until no further decrement of the objective function value is possible, i.e., $H_q\left(\boldsymbol{f}\left(\boldsymbol{z}_{j-1}\right)\right)=H_q\left(\boldsymbol{f}\left(\boldsymbol{z}_j\right)\right)$.

Notice that

$$\nabla_{\boldsymbol{x}_i}\big(H_q\left(\boldsymbol{f}\left(\boldsymbol{x}_1,\ldots,\boldsymbol{x}_N\right)\right)\big)=-\frac{1}{2N}\,\frac{q}{q-1}\boldsymbol{f}\left(\boldsymbol{x}_1,\ldots,\boldsymbol{x}_N\right)^{q-1},\ i\in\mathcal{I}, \tag{4.14}$$

where given $\boldsymbol{f}=\left[f_1\ldots f_M\right]^T$, we define $\boldsymbol{f}^{q-1}\triangleq\left[f_1^{q-1}\ldots f_M^{q-1}\right]^T$. Similarly as before, (4.13) can be decomposed into the following $|\mathcal{I}\,|$ integer optimiza-

tion problems, each solved with a binary integer linear program:

$$\min_{\boldsymbol{x}_i} \quad \left(-\frac{q}{q-1}\boldsymbol{f}\left(\boldsymbol{z}_{j-1}\right)^{q-1}\right)^T \left(\boldsymbol{x}_i - \boldsymbol{z}_{j-1}^i\right) \tag{4.15}$$

$$\text{subject to} \quad \begin{cases} \boldsymbol{g}_i = \boldsymbol{Z}\boldsymbol{x}_i \\ 2 = \mathbf{1}^T\boldsymbol{x}_i \\ \boldsymbol{x}_i \in \{0,1\}^M, \quad i \in \mathcal{I}, \end{cases}$$

where $\boldsymbol{z}_{j-1} = \left[\boldsymbol{z}_{j-1}^i,\ i \in \mathcal{I}\right]$.

Moreover, notice that for small q, the components of $\boldsymbol{f}\left(\boldsymbol{z}_{j-1}\right)$ that are zero will result in a solution $\tilde{\boldsymbol{x}}_i$ of (4.15) with zeroes in those same positions, in order to obtain a finite objective value. This is due to the fact that the components f_i that approach zero, have $f_i^{q-1} \to \infty$ for small $q < 1$. Furthermore, as we compute a list of all possible haplotype pairs for each observed genotype, we only need to consider the haplotype pairs that are in the list whose haplotypes do not correspond to null components of $\boldsymbol{f}\left(\boldsymbol{z}_{j-1}\right)$. This observation leads to a significant dimensionality reduction of (4.15).

4.3.2.3 Final Solution

The global and local searches need to be iterated until all local minimum points are visited. However, the number of such points is unknown a priori and a Bayesian estimate of the number of local minimum points is used instead [97]. In this way, with high probability, all local minima will be visited. Specifically, given the number w of observed local minima so far, and taking into account the estimate of the number local minima, a recommended

stopping time [97] is given by

$$(w^2 + w)/\delta + w + 2,$$

where δ is a parameter between zero and one. Notice that the stopping time needs to be updated every time a new local minimum is found and therefore, as more local minima are found, more iterations are carried out.

Let $\mathcal{K}$ be the final set of local minima visited. Then, the solution to the optimization problem (4.10) is given by the vertex in $\mathcal{K}$ that achieves the minimum value of the objective function, i.e.,

$$\{\hat{\boldsymbol{x}}_i,\ i \in \mathcal{I}\} = \arg \min_{\boldsymbol{x}_i,\ i \in \mathcal{I}} \{H_q\left(\boldsymbol{f}\left(\boldsymbol{x}_1 \ldots \boldsymbol{x}_N\right)\right) : \{\boldsymbol{x}_i,\ i \in \mathcal{I}\} \in \mathcal{K}\}.$$

Finally, we remark that the above approach to solving the sparse haplotyping problem (4.9) has the benefit of being amenable to parallel implementations. That is, starting with different random directions $\boldsymbol{u}$, the corresponding global and local searches can be performed in parallel. Moreover, the linear programs are decomposed into a set of $|\mathcal{I}\ |$ smaller linear programs which can also be solved in parallel. Note also that the parameter δ can be adjusted to tradeoff between speed and accuracy.

4.3.2.4 Summary of the Algorithm

Given the parameter $0 \leq \delta \leq 1$ that represents a trade-off between accuracy and running time, the algorithm proceeds as follows.

- Initialize the algorithm.

- Determine the set $\mathcal{I}$ of genotypes with two ambiguities or more. Find the list $\mathcal{L}_i$ of all possible haplotype pairs for each genotype in $\mathcal{I}$.
- Given the union of the lists $\mathcal{L}_i$ and the direct computations of the haplotypes associated with the genotypes not in $\mathcal{I}$, place each haplotype as a column of $\boldsymbol{Z}$.
- For the genotypes with no ambiguities, determine $\boldsymbol{x}_i$ by searching the column $\boldsymbol{g}_i/2$ of $\boldsymbol{Z}$.
- For the genotypes with only one ambiguity, set $h_i^1(\ell) = 0$ and $h_i^2(\ell) = 1$ where $g_i(\ell) = 1$, and $h_i^1(\ell) = h_i^2(\ell) = g_i(\ell)/2$ otherwise. Find the columns $\boldsymbol{h}_i^1$ and $\boldsymbol{h}_i^2$ of $\boldsymbol{Z}$.
- Start with an empty set of local minima, i.e., $\mathcal{K} = \emptyset$ and therefore, $w = 0$.

- Find the local minima.

  Starting with $j = 0$ and while $j < (w^2 + w)/\delta + w + 2$:

  - Find a vertex randomly using the global search.
    - Pick a random haplotype pair from $\mathcal{L}_i$ $\forall i \in \mathcal{I}$.
  - Find a local minimum through the local search step.
    - Stack the solutions to the global search $\hat{\boldsymbol{x}}_i, \forall i \in \mathcal{I}$, in $\boldsymbol{z}_0$.
    - Find the local minimum.

      Starting with $k = 1$, repeat until $H_q(\boldsymbol{z}_k) = H_q(\boldsymbol{z}_{k-1})$.

      · Solve (4.15) $\forall i \in \mathcal{I}$.

· Stack the solution in $\boldsymbol{z}_{k+1}$.

· $k \leftarrow k+1$.

– Let $\boldsymbol{z}_f$ be the output of the local search. If $\boldsymbol{z}_f \notin \mathcal{K}$, then it is the first time we visit this local minimum and we set $\mathcal{K} \leftarrow \mathcal{K} \cup \{\boldsymbol{z}_f\}$ and $w \leftarrow w+1$.

– $j \leftarrow j+1$.

- Find the global optimum point among the local minima.

$$\{\hat{\boldsymbol{x}}_i,\ i \in \mathcal{I}\} = \arg \min_{\boldsymbol{x}_i,\, i \in \mathcal{I}} \{H_q\left(\boldsymbol{f}\left(\boldsymbol{x}_1 \dots \boldsymbol{x}_N\right)\right) : \{\boldsymbol{x}_i,\ i \in \mathcal{I}\} \in \mathcal{K}\}.$$

4.4 Sparse Haplotyping based on Dictionary Selection

In the previous section we looked for a haplotype frequency vector that is as sparse as possible. The non-zero positions of the frequency vector correspond to columns in the matrix $\boldsymbol{Z}$ that are used to explain the genotypes. Therefore, searching for the sparsest frequency vector is equivalent to looking for the smallest set of columns of $\boldsymbol{Z}$ that explains all the observed genotypes. Each genotype can be reconstructed by one or two columns of the matrix $\boldsymbol{Z}$. But the maximum parsimony principle states that we should be able to reconstruct all genotypes using as few columns as possible, i.e., we look for a dictionary that can represent the genotypes using the least number of columns of $\boldsymbol{Z}$. In this section we state this approach mathematically and show how this is an approximately submodular optimization problem that can be solved efficiently with a greedy algorithm.

4.4.1 Problem Formulation

We denote the p-th column of $\boldsymbol{Z}$ as $\boldsymbol{h}^p$ and define the dictionary $\mathcal{D}$ as the indices of the columns of $\boldsymbol{Z}$ that are used to explain the observed genotypes. The maximum parsimony principle then dictates that the dictionary $\mathcal{D}$ should have the smallest possible cardinality. Let $\mathcal{A}_i \subseteq \mathcal{D}$ be the subset of the dictionary used to explain the genotype of the i-th person. If there is no ambiguity in $\boldsymbol{g}_i$, then $\mathcal{A}_i = \{m\}$ consists of only one index, corresponding to the m-th column $\boldsymbol{h}^m$ in matrix $\boldsymbol{Z}$ such that $\boldsymbol{g}_i = \boldsymbol{h}^m + \boldsymbol{h}^m$. On the other hand, when the genotype contains ambiguity in at least one locus, then two different haplotypes are needed to reconstruct the genotype and therefore, $\mathcal{A}_i = \{k, j\}$ consists of two indices such that $\boldsymbol{g}_i = \boldsymbol{h}^k + \boldsymbol{h}^j$.

Define the following coordinate vector of the i-th genotype

$$\tilde{\boldsymbol{x}}_i = \begin{cases} 2 & \text{if } i \notin \mathcal{I} \text{ and the genotype has no ambiguities,} \\ \begin{bmatrix} 1 & 1 \end{bmatrix}^T & \text{if } i \notin \mathcal{I} \text{ and the genotype has only one ambiguity,} \\ \begin{bmatrix} 1 & 1 \end{bmatrix}^T & \text{if } i \in \mathcal{I}. \end{cases} \tag{4.16}$$

Moreover, let $\boldsymbol{Z}_{\mathcal{A}_i}$ be the matrix formed by the columns of matrix $\boldsymbol{Z}$ indexed by $\mathcal{A}_i$. For each individual i, we then have

$$\boldsymbol{g}_i = \boldsymbol{Z}\boldsymbol{x}_i = \boldsymbol{Z}_{\mathcal{A}_i}\, \tilde{\boldsymbol{x}}_i. \tag{4.17}$$

Notice the connection between the vectors $\boldsymbol{x}_i$ and $\tilde{\boldsymbol{x}}_i$. The latter consists of all the non-zero components of the former; while the matrix $\boldsymbol{Z}_{\mathcal{A}_i}$ consists of the columns of matrix $\boldsymbol{Z}$ where $\boldsymbol{x}_i$ is non-zero.

The set $\{\mathcal{A}_i : i \notin \mathcal{I}\}$ can be easily found by searching through the columns of matrix $\boldsymbol{Z}$ as in the previous method. On the other hand, both $\{\mathcal{A}_i,\ i \in \mathcal{I}\}$ and $\mathcal{D}$ are unknown. The maximum parsimony haplotype inference problem is solved once we determine $\{\mathcal{A}_i,\ i \in \mathcal{I}\}$ with the cardinality of $\mathcal{D}$ as small as possible. Noticing that the cardinality of $\mathcal{A}_i$, i.e., $|\mathcal{A}_i|$, is always less than or equal to 2, the sparse haplotyping problem based on dictionary selection can then be summarized as follows.

Find $\mathcal{D}$ and $\{\mathcal{A}_i : |\mathcal{A}_i| \leq 2, \mathcal{A}_i \subseteq \mathcal{D}, i = 1, \ldots, N\}$ such that the cardinality of $\mathcal{D}$ is as small as possible and $\boldsymbol{g}_i = \boldsymbol{Z}_{\mathcal{A}_i}\tilde{\boldsymbol{x}}_i$, $i = 1, \ldots, N$.

4.4.2 Solution

The sparse haplotyping based on dictionary selection is an optimization problem that aims at solving the dictionary selection problem jointly with the reconstruction of the genotypes. This problem is combinatorial both in the selection of each $\mathcal{A}_i$ from the set $\mathcal{D}$ and in the selection of $\mathcal{D}$ from the available set of columns of $\boldsymbol{Z}$. In this subsection, we show that the joint optimization problem is approximately submodular and therefore, the haplotyping problem can be carried out with a low-complexity greedy method.

We first determine the set $\{\mathcal{A}_i : i \notin \mathcal{I}\}$ by searching through the columns of matrix $\boldsymbol{Z}$ as in the previous method. Then, to find the haplotypes for the subjects belonging to the set $\mathcal{I}$, we let $\tilde{\boldsymbol{x}}_i \in \mathbb{R}^2$, and define the following variance reduction metric over the set of observations:

$$L_i(\mathcal{A}) \quad = \quad \min_{\tilde{\boldsymbol{x}}_i} \|\boldsymbol{g}_i - \boldsymbol{Z}_{\mathcal{A}}\ \tilde{\boldsymbol{x}}_i\|^2, \tag{4.18}$$

where $\mathcal{A}$ is not yet determined. Notice that the final solution $\mathcal{D}$ and $\{\mathcal{A}_i : |\mathcal{A}_i| \leq 2, \mathcal{A}_i \subseteq \mathcal{D}, i \in \mathcal{I}\}$ will satisfy $L_i(\mathcal{A}_i) = 0$. Since a genotype should be explained by at most two columns of the matrix $\boldsymbol{Z}$, we can constrain the cardinality of $\mathcal{A}$ to be less than or equal to 2 and look for the set of columns of $\boldsymbol{Z}$ such that

$$\mathcal{A}_i \quad = \quad \arg \min_{\mathcal{A} \subseteq \mathcal{D}, |\mathcal{A}| \leq 2} L_i(\mathcal{A}), \tag{4.19}$$

where $\mathcal{D}$ is not yet determined. Notice that the cost function in (4.19) is zero when $\mathcal{D}$ contains a pair of haplotypes that explains the i-th genotype. Furthermore, we define

$$F_i(\mathcal{D}) \quad = \quad L_i(\emptyset) - \min_{\mathcal{A} \subseteq \mathcal{D}, |\mathcal{A}| \leq 2} L_i(\mathcal{A}), \tag{4.20}$$

with $L_i(\emptyset) = \|\boldsymbol{g}_i\|^2$ in order to have $F_i(\emptyset) = 0$. And to take into account all genotypes at once, we consider the average of all individuals as

$$F(\mathcal{D}) = \frac{1}{|\mathcal{I}|} \sum_{i \in \mathcal{I}} F_i(\mathcal{D}). \tag{4.21}$$

We then define the best dictionary of cardinality n as

$$\mathcal{D}_n^* = \arg \max_{|\mathcal{D}| \leq n} F(\mathcal{D}). \tag{4.22}$$

Therefore, the dictionary with the smallest possible cardinality that explains

all observed genotype is given by

$$\mathcal{D}^* = \min_n \left\{ \mathcal{D}_n^* : F(\mathcal{D}_n^*) = \frac{1}{|\mathcal{I}|} \sum_{i \in \mathcal{I}} \|\boldsymbol{g}_i\|^2 \right\}. \tag{4.23}$$

Given a cardinality n, the general setting of (4.22) was first considered in [99] in the context of synthetic signals and natural images for representation and inpainting problems. We extend the approach presented there in order to find the dictionary with the smallest possible cardinality of (4.23).

It is shown in [99] that (4.22) is monotonic, i.e., $F(\emptyset) = 0$ and whenever $\mathcal{D} \subseteq \mathcal{D}'$ then $F(\mathcal{D}) \leq F(\mathcal{D}')$, and approximately submodular with constant ϵ, i.e., for $\mathcal{D} \subseteq \mathcal{D}' \subseteq \mathcal{V}$ and $v \in \mathcal{V} \setminus \mathcal{D}'$ it holds that

$$F(\mathcal{D} \cup \{v\}) - F(\mathcal{D}) \geq F(\mathcal{D}' \cup \{v\}) - F(\mathcal{D}') - \epsilon,$$

where ϵ is related to the incoherence[2] of the column vectors of $\boldsymbol{Z}$.

The maximization of a monotonic and approximately submodular function can be approximately solved efficiently by a greedy algorithm with a convergence guarantee. A particular case is the maximization of a submodular function G, i.e., an approximately submodular function with $\epsilon = 0$. When applying a greedy technique that starts with the empty set and adds elements one by one, corresponding to the ones with the maximum marginal gains, we have the following result.

[2]The incoherence μ of a set of vectors of unit l_2 norm $\{\boldsymbol{\phi}_1, \ldots, \boldsymbol{\phi}_M\}$ is defined as

$$\mu \quad = \quad \max_{i \neq j} | < \boldsymbol{\phi}_i, \boldsymbol{\phi}_j > |.$$

Proposition 1 *[100] For a non-negative, monotone submodular function G, let $\hat{\mathcal{D}}_n$ be a set of size n obtained by selecting elements one at a time, each time choosing an element that provides the largest marginal increase in the function value. Then $G(\hat{\mathcal{D}}_n) \geq (1 - e^{-1}) \max_{\mathcal{D}:\, |\mathcal{D}| \leq n} G(\mathcal{D})$, where $e \approx 2.72$ is Euler's number.*

The theorem states that a greedy approach for solving the sparse dictionary selection problem provides a $(1 - e^{-1}) \approx 63\%$ approximation. When dealing with an approximately submodular function, the greedy solution also has the following property [101]:

$$F(\hat{\mathcal{D}}_n) \geq (1 - e^{-1}) \max_{\mathcal{D}:\, |\mathcal{D}| \leq n} F(\mathcal{D}) - n\epsilon$$

Therefore, we propose to solve the haplotype inference based on dictionary selection using a simple greedy algorithm with the above convergence property. The algorithm proceeds as follows.

Start with the empty set $\mathcal{D}_0 = \emptyset$, and at every iteration l, add the element m to $\mathcal{D}_{l-1}$ if the column vector $\boldsymbol{h}^m$ of $\boldsymbol{Z}$ is the one achieving the maximal marginal gain among the possible columns $1 \dots M$, that are not in $\mathcal{D}_{l-1}$, i.e.,

$$m = \arg \max_{k \in \{1 \dots M\} \setminus \mathcal{D}_{l-1}} F(\mathcal{D}_{l-1} \cup \{k\}),$$

until $F(\mathcal{D}_l) = \frac{1}{|\mathcal{I}|} \sum_{i \in \mathcal{I}} \|\boldsymbol{g}_i\|^2$.

We finally need to specify how to solve (4.18) and (4.19) within this greedy method. In our particular case, we know that for the i-th genotype

to be explained with a dictionary $\mathcal{D}$ when $i \in \mathcal{I}$, the vector $\tilde{\boldsymbol{x}}_i$ of (4.18) must equal $[1\ 1]^T$. This simplifies (4.18) as $\tilde{\boldsymbol{x}}_i$ is now fixed. We then find $\mathcal{A}_i$ in (4.19) by computing the difference of each genotype with the sum of all par of columns of $\boldsymbol{Z}_{\mathcal{D}}$, and pick the pair of columns that minimize (4.19). If the metric (4.18) is zero, it means that we can reconstruct the genotype with those two columns.

4.4.2.1 Summary of the Algorithm

- Initialize the algorithm.
  - Determine the set $\mathcal{I}$ of genotypes with two ambiguities or more.
  - Determine $\boldsymbol{Z}$ as in the previous method.
  - For the genotypes with no ambiguities determine $\boldsymbol{x}_i$ by searching the column $\boldsymbol{g}_i/2$ of $\boldsymbol{Z}$.
  - For the genotypes with only one ambiguity, set $h_i^1(\ell) = 0$ and $h_i^2(\ell) = 1$ where $g_i(\ell) = 1$, and $h_i^1(\ell) = h_i^2(\ell) = g_i(\ell)/2$ otherwise. Find the columns $\boldsymbol{h}_i^1$ and $\boldsymbol{h}_i^2$ of $\boldsymbol{Z}$.
  - Set the dictionary $\mathcal{D}_{n-1}^*$ to contain the indices of the columns of $\boldsymbol{Z}$ that explain the genotypes with no ambiguities and the genotypes with only one ambiguity, where $n-1$ is its cardinality.
- Iterate until all genotypes are explained, i.e., $F(\mathcal{D}_n^*) = \frac{1}{|\mathcal{I}|}\sum_{i\in\mathcal{I}} \|\boldsymbol{g}_i\|^2$.
  - Perform the greedy search.
    - For $\forall j \in \{1,\ldots,M\} \setminus \mathcal{D}_{n-1}^*$, compute $F(\mathcal{D}_{n-1}^* \cup \{j\})$ with $\tilde{\boldsymbol{x}}_i = [1\ 1]^T$ in (4.18).

* Let $j^* = \arg\max_{j \in \{1 \ldots M\} \setminus \mathcal{D}^*_{n-1}} F(\mathcal{D}^*_{n-1} \cup \{j\})$. Set $\mathcal{D}^*_n = \mathcal{D}^*_{n-1} \cup \{j^*\}$.
* Check if any genotype is explained by the addition of the new element $\boldsymbol{h}^{j^*}$, i.e., if (4.18) is zero. If so, the inferred haplotype pair for the individual with such a genotype is $[\boldsymbol{h}^{j^*},\ \boldsymbol{g}_k - \boldsymbol{h}^{j^*}]$.

– $n \leftarrow n + 1$.

4.5 Extensions

4.5.1 Large Data Sets

When the number of ambiguous sites is large, the complexity of finding the matrix $\boldsymbol{Z}$ increases dramatically. One approach for this case is to partition the data into blocks and process one block at a time. After all blocks are processed, a ligation process is performed to obtain the final result. We next adapt such a partition-ligation (PL) method [86] to the sparse haplotyping approach.

The PL method starts with the partition phase. The genotype data is divided into Q non-overlapping and non-empty sets that cover all of the genotypes. Each set contains genotype segments from the same SNP loci for all individuals. Let $\left\{\boldsymbol{G}_{q_1^1:q_2^1}, \boldsymbol{G}_{q_1^2:q_2^2} \ldots, \boldsymbol{G}_{q_1^Q:q_2^Q}\right\}$ be the partitioned sets of genotype data, where the i-th subset $\boldsymbol{G}_{q_1^i:q_2^i}$ contains the genotypes for SNP locus q_1^i to q_2^i for all N individuals. We impose that the first locus of the first set be the first locus of the complete genotype, i.e., $q_1^1 = 1$. Moreover, each set is adjacent to the previous one, i.e., $q_1^i = q_2^{i-1} + 1$ for $i = \{2 \ldots Q\}$. Notice that as we need to cover all loci, the last locus for the

last set is $q_2^Q = L$. For each set $\boldsymbol{G}_{q_1^i:q_2^i}$, the haplotypes are inferred using our algorithm, which outputs a small set of haplotypes $\left\{\boldsymbol{h}_1^i \ldots \boldsymbol{h}_{K_i}^i\right\}$ that can be used to explain the set of genotypes, where K_i is the number of haplotypes in the set.

Then, the PL proceeds to a ligation phase, where adjacent sets are merged to obtain a new partition of the data, with $\left\lceil \frac{Q}{2} \right\rceil$ sets, e.g., when merging the $(2i)$-th set with the $(2i+1)$-th set, the resulting set consists of the genotypes for all individuals between locus q_1^{2i} and q_2^{2i+1}. For each merged set $\boldsymbol{G}_{q_1^{2i}:q_2^{2i+1}}$, we run the haplotype inference algorithm again, but restricting $\boldsymbol{Z}$ to contain every possible concatenations of the K_{2i} haplotypes of the $(2i)$-th set with the K_{2i+1} haplotypes of the $(2i+1)$-th set. The process continues until there is only one set of genotypes and the haplotype inference algorithm is finally applied to this set.

In order to use the PL method, we need to determine an initial partition of the data. Therefore, we need to specify the number of partitions Q and the length of each partition or equivalently, the initial locus of each partition, i.e., $\left\{q_1^i\right\}_{i=1\ldots Q}$. A simple and low-cost way of setting the initial loci $\left\{q_1^i\right\}_{i=1\ldots Q}$ is to fix each block to be of equal length. Then, given an upper bound W on the length for each initial block, the number of blocks is $Q = \lceil \frac{L}{W} \rceil$.

Another option to initialize the PL method is to perform block partitioning. In this case, Q and $\left\{q_1^i\right\}_{i=1\ldots Q}$ are chosen in order to take advantage of the block structure that the haplotypes naturally exhibit between recombination hot-spots [92]. Within hot-spots, the haplotype fragments display less diversity compared to the complete haplotype vectors. We measure the

diversity in a block by computing its Shannon entropy. Let $\left\{\tilde{\boldsymbol{h}}_1^{ij} \dots \tilde{\boldsymbol{h}}_{\tilde{K}_{ij}}^{ij}\right\}$ be the $\tilde{K}_{ij}$ haplotypes used to generate the genotypes in the block that start at locus i and finishes at locus j, and $\tilde{\boldsymbol{f}}^{ij} = \left[\tilde{f}_1^{ij}, \dots, \tilde{f}_{\tilde{K}_{ij}}^{ij}\right]$ the haplotype frequency vector. Then, the entropy of the this block is given by

$$E(i,j) \quad = \quad -\sum_{k=1}^{\tilde{K}_{ij}} \tilde{f}_k^{ij} \log \tilde{f}_k^{ij}. \tag{4.24}$$

Moreover, we define the total entropy as the sum of the entropies for the haplotype segments of each block. We look for the best partitioning by finding the blocks such that the total entropy is minimized. However, the true haplotypes are not known, but given $\boldsymbol{G}_{i:j}$, they can be inferred with either the sparse haplotyping based on Tsallis entropy minimization method or the sparse haplotyping based on dictionary selection approach. Let $C(k)$ be the minimum total block entropy up to k-th SNP, and $E(i,j)$ the entropy of the frequencies of the haplotype pairs. Then, we need to compute $C(L)$ and we do so by solving the following recursive problem. Set $C(0) = 0$ and, for $k = 1, \dots, L$, compute

$$C(k) \quad = \quad \min_{1 \le i \le k} \left\{C(i-1) + E(i,k); \quad \text{for } k - i \le W\right\}, \tag{4.25}$$

where W is the upper bound for the length of the block. Notice that the argument i in (4.25) gives the optimal initial point of the block ending in locus k. Therefore, after computing $C(L)$, the minimum total entropy for all blocks, we need to do backtracking in order to uncover the optimal block partitioning. For example, once $C(L)$ is computed, we look at the i^* in

(4.25) that was used to find the minimum. Such i^* defines the initial locus of the last block. We then look at $C(i^* - 1)$ and the argument i that was used to achieve the minimum, and so on until we reach the initial locus.

4.5.2 Missing Data

Errors often occur during the genotyping process, and the data at some loci in some genotypes might not have been observed. We present modifications to the algorithms to perform haplotype inference in the presence of missing data. We assume that it is known a priori where the genotype information is missing for each genotype of each individual.

Missing data can occur both in genotypes in $\mathcal{I}$ and in genotypes not in $\mathcal{I}$. In the latter case, there is uncertainty regarding the haplotype pair of genotypes that originally had no ambiguity. We need to expand the set $\mathcal{I}$ in order to take this into account. Let $\mathcal{M}$ be the set of indices of genotypes with missing information. The indices of genotypes that present uncertainty is then given by $\mathcal{J} = \mathcal{M} \cup \mathcal{I}$.

We next present the modifications to both the sparse haplotyping based on Tsallis entropy minimization and the sparse haplotyping based on dictionary selection in order to handle missing data.

4.5.2.1 Sparse Haplotyping based on Tsallis Entropy Minimization

The observed genotypes define the constraints of the minimization problem in (4.9). Missing data then implies a smaller number of constraints. Let $\tilde{\boldsymbol{g}}_i$ be the genotype $\boldsymbol{g}_i$ where all the loci with missing information have been

removed, and $\tilde{\boldsymbol{Z}}^i$ the matrix $\boldsymbol{Z}$ with all the rows corresponding to those loci removed. Notice that different individuals present missing information in different loci, making the matrix $\tilde{\boldsymbol{Z}}^i$ dependant on the considered individual. The solution to the sparse haplotyping based on Tsallis entropy minimization then needs to satisfy $\tilde{\boldsymbol{g}}_i = \tilde{\boldsymbol{Z}}^i \boldsymbol{x}_i, \ \forall i \in \mathcal{J}$, and the sparse haplotyping based on Tsallis entropy minimization becomes

$$\begin{aligned} \min_{\boldsymbol{x}_i,\, i\in\mathcal{J}} \quad & H_q\left(\boldsymbol{f}(\boldsymbol{x}_1,\ldots,\boldsymbol{x}_N)\right) \\ \text{subject to} \quad & \begin{cases} \tilde{\boldsymbol{g}}_i = \tilde{\boldsymbol{Z}}^i \boldsymbol{x}_i \\ 2 = \mathbf{1}^T \boldsymbol{x}_i \\ \boldsymbol{x}_i \in \{0,1,2\}^M, \quad i \in \mathcal{J}. \end{cases} \end{aligned} \tag{4.26}$$

The solution to this problem can be found following an analogous procedure to the one given in Section 4.3.

4.5.2.2 Sparse Haplotyping based on Dictionary Selection

This method is based on the idea of finding the least number of columns of the matrix $\boldsymbol{Z}$ to explain all the genotypes. Moreover, removing the loci of missing information from a genotype implies removing the corresponding rows of matrix $\boldsymbol{Z}$. Therefore, it is still valid to look for the minimum set of columns that explain the genotypes. We then modify (4.18) as follows.

$$L_i(\mathcal{A}) \quad = \quad f(dim(\tilde{\boldsymbol{g}}_i)) \min_{\tilde{\boldsymbol{x}}_i} \|\tilde{\boldsymbol{g}}_i - \tilde{\boldsymbol{Z}}^i_{\mathcal{A}}\, \tilde{\boldsymbol{x}}_i\|^2, \tag{4.27}$$

where $dim(\boldsymbol{a})$ gives the dimension of vector $\boldsymbol{a}$, and $f(.) : \mathbb{N} \rightarrow \mathbb{R}$. We want to give more weight to genotypes with less missing observations as they contain more information, so we restrict $f(.)$ to be nondecreasing. We found experimentally that setting $f(dim(\boldsymbol{a}))) = dim(\boldsymbol{a})^2$ achieves a good performance. The greedy algorithm given in Section 4.4 needs to be modified accordingly.

4.6 Experimental Results

To assess the performance of the different algorithms, we use two different metrics. The first one is the error rate P_{error} which is the proportion of individuals whose haplotypes are incorrectly inferred. This measure however does not give an idea of how different an incorrect pair of estimated haplotypes is when compared with the true pair of haplotypes. For that, we use a second measure of performance given by the switch error rate [76].

The switch error rate measures how dissimilar the inferred pair of haplotypes is with respect to the true pair for each individual. This measure only takes into account heterozygous sites, as errors can only happen in this type of sites (the homozygous sites are fully determined by the genotypes). Let $[\boldsymbol{h}^1, \boldsymbol{h}^2]$ be the true haplotypes and $[\hat{\boldsymbol{h}}^1, \hat{\boldsymbol{h}}^2]$ the inferred ones. Notice that $\hat{\boldsymbol{h}}^1$ is not necessary an estimate of $\boldsymbol{h}^1$ as it can be an estimate of $\boldsymbol{h}^2$. Moreover, notice that for every heterozygous site ℓ we have $h^1(\ell) + h^2(\ell) = 1$, i.e., $h^1(\ell)$ is 0 and $h^2(\ell)$ is 1 or viceversa. The same occurs with $\hat{h}^1(\ell)$ and $\hat{h}^2(\ell)$. We define a *switch* as the flip between the 0 and the 1 in a particular locus for the pair of haplotypes. Given a genotype with s_i heterozygous sites,

one needs at most $\left\lfloor \frac{s_i}{2} \right\rfloor$ switches to transform the inferred haplotypes to the true ones. Moreover, in a database of N individuals, the total number of switches in the worst case scenario is

$$\sum_{i=1}^{N} \left\lfloor \frac{s_i}{2} \right\rfloor . \tag{4.28}$$

The switch error rate is then the ratio between the actual number of switches that are needed to go from the inferred haplotypes to the true ones and the worst case number of switches of (4.28).

We apply the two proposed methods in this chapter to infer the haplotypes given genotypes from three different data sets. The first one consists of synthetic data generated using the coalescence theory. The second one corresponds to the Angiotensin Converting Enzyme (ACE) data set that is considered to be a data set with a number of subjects not large enough in comparison with the number of distinct haplotypes [76]. And finally, we test the algorithm with the Cystic Fibrosis Transmembrane-Conductance Regulator (CFTR) Gene data set where some haplotypes are only used once to generate the observed genotypes.

The performance of the two proposed methods are compared with four state-of-the-art haplotype inference methods. In particular, we present the results of applying PHASE [81] and its fast version fastPHASE [90], Gerbil [89] and CollHaps [76] to the same data sets.

4.6.1 Synthetic Data

Haplotypes are simulated and randomly paired to form genotypes of short sequence data (between 5 and 30 SNPs in each haplotype). An infinite-site model [102] with $\theta = 4$ and a recombination rate $r \in \{0, 4, 40\}$ was used in the coalescence-based program of R.R. Hudson. For each value of r, 100 data sets were generated for different numbers of individuals in the set (10 to 50 individuals in each data set). Figure 4.2 shows the maximum and mean number of ambiguous sites in each genotype in the 100 data sets for each different number of individuals. The minimum is not shown as in it zero, that is, there is at least one genotype with no ambiguity in one of the datasets. Moreover, the resulting max, mean and min number of different haplotypes used in each data set is shown.

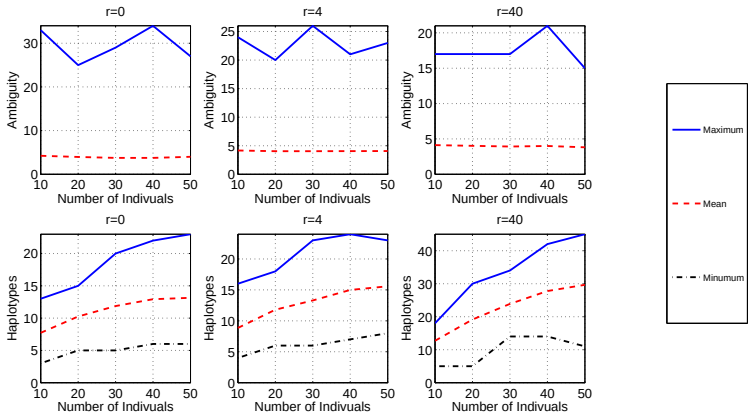


Figure 4.2: The number of ambiguous sites and the number of haplotypes used in each dataset.

Both proposed methods are applied to this data sets, i.e., the sparse haplotyping based on Tsallis entropy minimization (SHTEM) with $q = 0.01$

and the sparse haplotyping based on dictionary selection (SHDS). The PL method is used when the genotypes are longer than 15 SNPs, by dividing the genotypes into two fixed blocks of length equal to half the length of the genotype vectors.

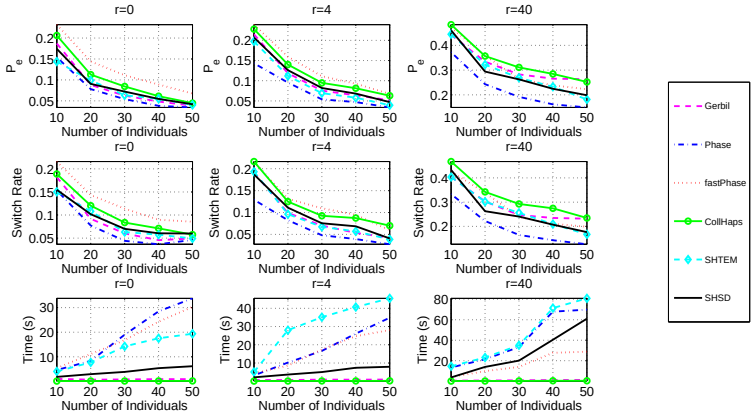


Figure 4.3: Probability of error, switch rate and average running time for the synthetic database.

Figure 4.3 shows the average of both measures of performance for different values of recombination rate and number of individuals over each of the 100 data sets. It is seen that PHASE is the method performing the best in almost all databases, with SHTEM having a comparable performance. Both of these good performances are achieved at the expense of high computational complexity. On the other hand, among the set of fast methods, i.e., SHDS, fastPHASE, Gerbil and CollHaps, SHDS is the method outperforming the others for almost all scenarios.

Both SHTEM and SHDS are implemented in Matlab. The other methods were obtained from the authors' websites: Gervil is implemented in Java,

	Number of Haplotypes	Genotype Error Rate	Switch Error Rate
Gerbil	13	0.1818	0.0047
fastPhase	13	0.1818	0.0047
Phase	13	0.1818	0.0063
CollHaps	13	0.2727	0.0058
SHTEM	13	0.1818	0.0063
SHSD	13	0.1818	0.0035

Table 4.1: Performance of different methods over the ACE data set.

CollHaps, PHASE and fastPHASE in C++. Figure 4.3 shows the average running time per data set of the different methods. It is seen that the running time of SHTEM is in the order of the running time of PHASE, while SHSD is faster and closer in average running time of CollHaps and Gerbil.

4.6.2 Angiotensin Converting Enzyme Data Set

The angiotensin converting enzyme (ACE) plays an important role in the control of systemic blood pressure and fluid-electrolyte balance by catalysing the conversion of angiotensin I to peptide angiotensin II. In [103], the complete DNA sequence of the gene that encodes the enzyme is presented for 11 individuals. It was found that there are 13 distinct haplotypes that are needed to reconstruct the database. For each genotypes, a set of 52 SNPs is considered as this is the set of non-unique polymorphic sites. The genotypes have a maximum of 37 ambiguous sites and there is 1 genotype with no ambiguity. The mean number of sites with ambiguity is 17.91.

We tested both the SHTEM and SHSD methods. The genotypes are first partitioned using the PL method with initial blocks selected according to the recursive algorithm of (4.25). An upper bound for the length of the block $W = 8$ is used.

The results of applying the different methods to this database are shown in Table 4.1. All methods have the same genotype error rate except for CollHaps, which has a higher error rate. This poor result of CollHaps was expected, as is stressed in [76] the importance of using data sets with a large number of individuals in order to achieve a good performance by this method.

Moreover, it is seen from the table that SHSD is the haplotyping method achieving the lowest switch error rate. This means that among the algorithms achieving the same error rate, SHSD is the one finding the solution that is closest to the real solution.

4.6.3 Cystic Fibrosis Transmembrane-Conductance Regulator Gene Data Set

The gene that encodes the Cystic Fibrosis Transmembrane Conductance Regulator (CFTR) is related to cystic fibrosis and congenital absence of the vas deferens, as the protein it encodes transports chloride ions and thiocyanate across cell membranes. In [104], 29 distinct haplotypes containing 23 SNPs each with no missing data are given. We combined these haplotypes randomly to get the genotypes of N individuals and compared the performance of different methods on this data sets. For $N \in [100, 500]$, the resulting mean number of ambiguous sites per genotype is in the interval

[9.15, 9.18]. The maximum number of sites with ambiguity is 20, regardless of the value of N, and the minimum is 0.

This database is characterized by the fact that for small numbers of individuals N, many of the distinct haplotypes are only used once. Therefore, methods as PHASE that use more biological-meaningful models are expected to perform better.

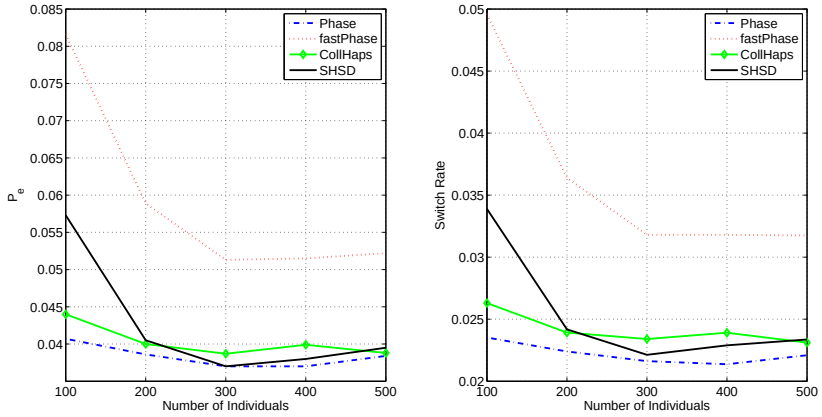


Figure 4.4: Probability of error and switch rate for the CFTR database.

We do not consider the SHTEM method as its complexity makes it unsuitable for data sets of this sizes. On the other hand, SHSD was applied with the PL method, with initial 4 blocks of equal size. The probability of error and the switch error rate are shown in Figure 4.4. It is seen that PHASE, being one of the methods that use more biological side-information, is the one achieving the lowest errors, but with a high algorithmic complexity. Moreover, both SHSD and CollHaps offer comparable performances with that of PHASE, despite being faster methods. The performance of Gerbil is not shown as its poor performance for this dataset is out of scale.

Moreover, it is seen that fastPHASE performs poorly in comparison to the equally fast methods SHSD and CollHaps.

4.6.4 Missing Data

As all algorithms have the same probability of error when considering the ACE database, we use this database to compare the performance when there is missing data in the input genotypes. For each of the 11 individuals in the dataset, we assume that each SNP has a probability P_{miss} of being marked as a missing SNP. Then, for each P_{miss}, we generated 100 realizations of the database and tested GERBIL, fastPHASE, PHASE, and SHSD as they are capable of handling missing data.

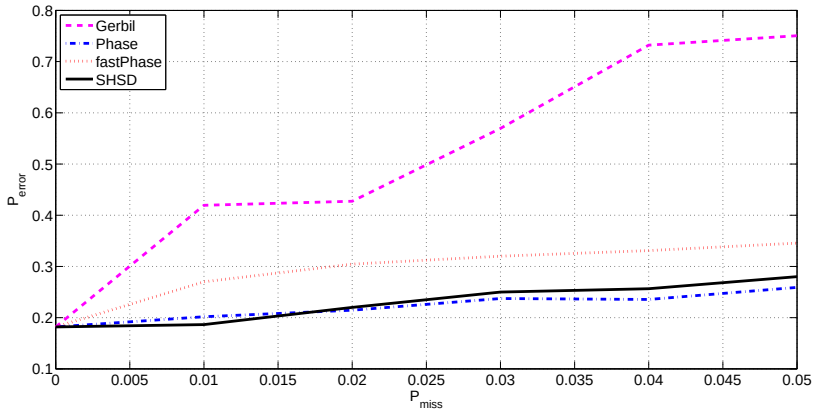


Figure 4.5: Probability of error versus probability of missing data in the ACE database.

SHSD is again used with the PL method, by first partitioning the genotypes according to the recursive algorithm of (4.25). An upper bound for the length of the block $W = 8$ is used.

It is seen in Figure 4.5 that, although SHSD has a lower complexity than

PHASE, it achieves a comparable probability of error and for some P_{miss}, it is the algorithm performing the best. Moreover, it is seen that this method is robust against missing data, as the slope of growth is smaller compared with Gerbil and fastPHASE.

Bibliography

[1] J. Watson and F. Crick, "Molecular structure of nucleic acids; a structure for deoxyribose nucleic acid," *Nature*, vol. 171, no. 4356, pp. 737–738, 1953.

[2] F. Crick, "On protein synthesis," in *Symposia of the Society for Experimental Biology*, vol. 12, 1958, pp. 138–163.

[3] F. Crick *et al.*, "Central dogma of molecular biology," *Nature*, vol. 227, no. 5258, pp. 561–563, 1970.

[4] F. Collins, E. Lander, J. Rogers, R. Waterston, and I. Conso, "Finishing the euchromatic sequence of the human genome," *Nature*, vol. 431, no. 7011, pp. 931–945, 2004.

[5] R. Karp, "Mathematical challenges from genomics and molecular biology," *Notices of the AMS*, vol. 49, no. 5, pp. 544–553, 2002.

[6] F. Sanger, S. Nicklen, and A. Coulson, "DNA sequencing with chain-terminating inhibitors," *Proceedings of the National Academy of Sciences*, vol. 74, no. 12, p. 5463, 1977.

[7] J. Shendure and H. Ji, "Next-generation DNA sequencing," *Nature biotechnology*, vol. 26, no. 10, pp. 1135–1145, 2008.

[8] M. Pop and S. Salzberg, "Bioinformatics challenges of new sequencing technology," *Trends in Genetics*, vol. 24, no. 3, pp. 142–149, 2008.

[9] E. Pettersson, J. Lundeberg, and A. Ahmadian, "Generations of sequencing technologies," *Genomics*, vol. 93, no. 2, pp. 105–111, 2009.

[10] A. Geller and M. O'Connor, "The sickle cell crisis: a dilemma in pain relief," *Mayo Clinic Proceedings*, vol. 83, no. 3, pp. 320–323, 2008.

[11] E. Korotkov and D. Phoenix, "Latent periodicity of DNA sequences of many genes," in *Proc. Pacific Symp. Biocomput.*, vol. 97, 1997, pp. 222–229.

[12] G. Jajamovich, A. Tajer, and X. Wang, "Minimax-optimal composite hypothesis testing," *IEEE Trans. on Signal Process., under review*, 2012.

[13] G. Jajamovich, X. Wang, A. Arkin, and M. Samoilov, "Bayesian multiple-instance motif discovery with bambi: inference of recombinase and transcription factor binding sites," *Nucleic Acids Research, doi:10.1093/nar/gkr745*, 2011.

[14] G. Jajamovich and X. Wang, "Maximum-parsimony haplotype inference based on sparse representations of genotypes," *IEEE Trans. on Signal Process., early access*, 2012.

[15] H. Poor, *An Introduction to Signal Detection and Estimation*. New York: Springer, 1994.

[16] O. Zeitouni, J. Ziv, and N. Merhav, "When is the generalized likelihood ratio test optimal?" *IEEE Trans. Inform. Theory*, vol. 38, no. 5, pp. 1597–1602, 1992.

[17] M. Feder and N. Merhav, "Universal composite hypothesis testing: A competitive minimax approach," *IEEE Trans. Inform. Theory*, vol. 48, no. 6, pp. 1504–1517, 2002.

[18] A. Wald, "Statistical decision functions," *Ann. Math. Statist.*, vol. 20, no. 2, pp. 165–205, 1949.

[19] D. Middleton and R. Esposito, "Simultaneous optimum detection and estimation of signals in noise," *IEEE Trans. Inform. Theory*, vol. 14, no. 3, pp. 434–444, 1968.

[20] A. Fredriksen, D. Middleton, and V. VandeLinde, "Simultaneous signal detection and estimation under multiple hypotheses," *IEEE Trans. Inform. Theory*, vol. 18, no. 5, pp. 607–614, 1972.

[21] G. Moustakides, "Finite sample size optimality of GLR tests," *Arxiv preprint arXiv:0903.3795*, 2009.

[22] B. Baygun and A. Hero III, "Optimal simultaneous detection and estimation under a false alarm constraint," *IEEE Trans. Inform. Theory*, vol. 41, no. 3, pp. 688–703, 1995.

[23] G. Moustakides, G. Jajamovich, A. Tajer, and X. Wang, "Joint detection and estimation: Optimum tests and applications," *IEEE Trans. Inform. Theory*, vol. PP, no. 99, p. 1, 2012.

[24] S. Boyd and L. Vandenberghe, *Convex optimization*. Cambridge, U.K.: Cambridge University Press, 2004.

[25] J. Font-Segura and X. Wang, "GLRT-based spectrum sensing for cognitive radio with prior information," *IEEE Trans. Commun.*, vol. 58, no. 7, pp. 2137–2146, july 2010.

[26] C. Hearne, S. Ghosh, and J. Todd, "Microsatellites for linkage analysis of genetic traits," *Trends in Genetics*, vol. 8, no. 8, pp. 288–294, 1992.

[27] E. Korotkov and N. Kudryaschov, "Latent periodicity of many genes," *Genome Informatics*, vol. 12, pp. 437–439, 2001.

[28] M. Chaley, E. Korotkov, and K. Skryabin, "Method revealing latent periodicity of the nucleotide sequences modified for a case of small samples," *DNA Research*, vol. 6, no. 3, pp. 153–163, 1999.

[29] J. Epps, H. Ying, and G. Huttley, "Statistical methods for detecting periodic fragments in DNA sequence data," *Biology Direct*, vol. 6, no. 21, pp. 1–16, 2011.

[30] D. Anastassiou, "Genomic signal processing," *IEEE Signal Process. Mag.*, vol. 18, no. 4, pp. 8–20, 2001.

[31] R. Arora and W. Sethares, "Detection of periodicities in gene sequences: a maximum likelihood approach," in *IEEE Int'l Workshop on Genomic Signal Processing and Statistics (GENSIP'07)*, 2007, pp. 1–4.

[32] R. Arora, W. Sethares, and J. Bucklew, "Latent periodicities in genome sequences," *IEEE J. Select. Topics Sig. Proc.*, vol. 2, no. 3, pp. 332–342, 2008.

[33] A. Lehninger, D. Nelson, and M. Cox, *Principles of Biochemistry*, 2nd ed. New York: Worth Publ., 1993.

[34] M. Tompa, N. Li, T. Bailey, G. Church, B. De Moor, E. Eskin, A. Favorov, M. Frith, Y. Fu, W. Kent, *et al.*, "Assessing computational tools for the discovery of transcription factor binding sites," *Nature biotechnology*, vol. 23, no. 1, pp. 137–144, 2005.

[35] T. Bailey, N. Williams, C. Misleh, and W. Li, "MEME: discovering and analyzing DNA and protein sequence motifs," *Nucleic acids research*, vol. 34, no. suppl 2, pp. W369–W373, 2006.

[36] D. Reiss, N. Baliga, and R. Bonneau, "Integrated biclustering of heterogeneous genome-wide datasets for the inference of global regulatory networks," *BMC bioinformatics*, vol. 7, no. 1, pp. 280–302, 2006.

[37] P. Pevzner and S. Sze, "Combinatorial approaches to finding subtle signals in DNA sequences," in *Proc. Int. C Int. Sys. Mol. Bio.*, vol. 8, 2000, pp. 269–278.

[38] G. Hertz and G. Stormo, "Identifying DNA and protein patterns with statistically significant alignments of multiple sequences," *Bioinformatics*, vol. 15, no. 7, pp. 563–577, 1999.

[39] S. Sinha and M. Tompa, "Ymf: A program for discovery of novel transcription factor binding sites by statistical overrepresentation," *Nucleic acids research*, vol. 31, no. 13, pp. 3586–3588, 2003.

[40] G. Pavesi, G. Mauri, and G. Pesole, "An algorithm for finding signals of unknown length in DNA sequences," *Bioinformatics*, vol. 17, no. suppl 1, pp. S207–S214, 2001.

[41] S. Sze and X. Zhao, "Improved pattern-driven algorithms for motif finding in DNA sequences," *Systems Biology and Regulatory Genomics*, vol. 4023, pp. 198–211, 2006.

[42] T. Bailey and C. Elkan, *Fitting a mixture model by expectation maximization to discover motifs in bipolymers*, ser. Technical Report. Dept. of Computer Science and Engineering, University of California, San Diego, 1994.

[43] T. Bailey, M. Boden, F. Buske, M. Frith, C. Grant, L. Clementi, J. Ren, W. Li, and W. Noble, "MEME SUITE: tools for motif discovery and searching," *Nucleic Acids Research*, vol. 37, no. suppl 2, pp. W202–W208, 2009.

[44] X. Liu, D. Brutlag, and J. Liu, "Bioprospector: discover conserved DNA motifs in upstream regulatory regions of co-expressed genes," in *Proc. 6th Pacific Symposium on Biocomputing*, 2001.

[45] J. Hughes, P. Estep, S. Tavazoie, and G. Church, "Computational identification of cis-regulatory elements associated with groups of functionally related genes in Saccharomyces cerevisiae," *Journal of molecular biology*, vol. 296, no. 5, pp. 1205–1214, 2000.

[46] J. Liu, A. Neuwald, and C. Lawrence, "Bayesian models for multiple local sequence alignment and gibbs sampling strategies," *Journal of the American Statistical Association*, pp. 1156–1170, 1995.

[47] G. Thijs, M. Lescot, K. Marchal, S. Rombauts, B. De Moor, P. Rouze, and Y. Moreau, "A higher-order background model improves the detection of promoter regulatory elements by gibbs sampling," *Bioinformatics*, vol. 17, no. 12, p. 1113, 2001.

[48] A. Favorov, M. Gelfand, A. Gerasimova, D. Ravcheev, A. Mironov, and V. Makeev, "A gibbs sampler for identification of symmetrically structured, spaced DNA motifs with improved estimation of the signal length," *Bioinformatics*, vol. 21, no. 10, pp. 2240–2245, 2005.

[49] K. Liang, X. Wang, and D. Anastassiou, "A profile-based deterministic sequential Monte Carlo algorithm for motif discovery," *Bioinformatics*, vol. 24, no. 1, pp. 46–55, 2008.

[50] S. Jensen and J. Liu, "BioOptimizer: a Bayesian scoring function approach to motif discovery," *Bioinformatics*, vol. 20, no. 10, pp. 1557–1564, 2004.

[51] A. Doucet, N. De Freitas, and N. Gordon, *Sequential Monte Carlo Methods in Practice*. New York: Springer Verlag, 2001.

[52] B. Dong, X. Wang, and A. Doucet, "A new class of soft MIMO demodulation algorithms," *IEEE Trans. Signal Process.*, vol. 51, no. 11, pp. 2752–2763, 2003.

[53] M. Evans, N. Hastings, and B. Peacock, *Statistical distributions*. New York: Wiley-Interscience, 2000.

[54] T. Vercauteren, D. Guo, and X. Wang, "Joint multiple target tracking and classification in collaborative sensor networks," *IEEE Journal on Selected Areas in Communications*, vol. 23, no. 4, pp. 714–723, 2005.

[55] M. Burset and R. Guigó, "Evaluation of gene structure prediction programs," *Genomics*, vol. 34, no. 3, pp. 353–367, 1996.

[56] A. Kolb, S. Busby, I. Buc, S. Garges, and S. Adhya, "Transcriptional regulation by camp and its receptor protein," *Annual review of biochemistry*, vol. 62, no. 1, pp. 749–797, 1993.

[57] I. Cases and V. de Lorenzo, "Expression systems and physiological control of promoter activity in bacteria," *Current opinion in microbiology*, vol. 1, no. 3, pp. 303–310, 1998.

[58] T. Desai, D. Rodionov, M. Gelfand, E. Alm, and C. Rao, "Engineering transcription factors with novel DNA-binding specificity using comparative genomics," *Nucleic acids research*, vol. 37, no. 8, pp. 2493–2503, 2009.

[59] G. Stormo and G. Hartzell, "Identifying protein-binding sites from unaligned DNA fragments," *Proceedings of the National Academy of Sciences*, vol. 86, no. 4, p. 1183, 1989.

[60] N. Grindley, K. Whiteson, and P. Rice, "Mechanisms of Site-Specific Recombination," *Annual Review of Biochemistry*, vol. 75, pp. 567–605, 2006.

[61] H. Kuwahara, C. Myers, and M. Samoilov, "Temperature control of fimbriation circuit switch in uropathogenic escherichia coli: quantitative analysis via automated model abstraction," *PLoS Computational Biology*, vol. 6, no. 3, p. e1000723, 2010.

[62] R. Johnson, "Bacterial site-specific DNA inversion systems," in *Mobile DNA II*. American Society for Microbiology Press, 2002, ch. 13, pp. 230–271.

[63] M. Smith and H. Thorpe, "Diversity in the serine recombinases," *Molecular microbiology*, vol. 44, no. 2, pp. 299–307, 2002.

[64] E. Moses, R. Good, M. Sinistaj, S. Billington, C. Langford, and J. Rood, "A multiple site-specific DNA-inversion model for the control of ompi phase and antigenic variation in dichelobacter nodosus," *Molecular microbiology*, vol. 17, no. 1, pp. 183–196, 1995.

[65] A. Tominaga, personal communication, 2010.

[66] H. Sandmeier, S. Iida, J. Meyer, R. Hiestand-Nauer, and W. Arber, "Site-specific DNA recombination system min of plasmid p15b: a cluster of overlapping invertible DNA segments," *Proceedings of the National Academy of Sciences*, vol. 87, no. 3, p. 1109, 1990.

[67] P. Crellin and J. Rood, "The resolvase/invertase domain of the site-specific recombinase tnpx is functional and recognizes a target sequence that resembles the junction of the circular form of the clostridium perfringens transposon tn4451," *Journal of bacteriology*, vol. 179, no. 16, p. 5148, 1997.

[68] A. Tominaga, S. Ikemizu, and M. Enomoto, "Site-specific recombinase genes in three shigella subgroups and nucleotide sequences of a pinb gene and an invertible b segment from shigella boydii." *Journal of bacteriology*, vol. 173, no. 13, p. 4079, 1991.

[69] H. Sandmeler, "Acquisition and rearrangement of sequence motifs in the evolution of bacteriophage tail fibres," *Molecular microbiology*, vol. 12, no. 3, pp. 343–350, 1994.

[70] M. Hoehe, K. Köpke, B. Wendel, K. Rohde, C. Flachmeier, K. Kidd, W. Berrettini, and G. Church, "Sequence variability and candidate gene analysis in complex disease: association of μ opioid receptor gene variation with substance dependence," *Human Molecular Genetics*, vol. 9, no. 19, pp. 2895–2908, 2000.

[71] L. Jin, P. Underhill, V. Doctor, R. Davis, P. Shen, L. Cavalli-Sforza, and P. Oefner, "Distribution of haplotypes from a chromosome 21 region distinguishes multiple prehistoric human migrations," *Proceedings of the National Academy of Sciences*, vol. 96, no. 7, pp. 3796–3800, 1999.

[72] S. Aluru, *Handbook of Computational Molecular Biology*. Chapman & Hall/CRC, 2005.

[73] E. Hubbell, "Finding a parsimony solution to haplotype phase is NP-hard," Personal communication, 2002.

[74] G. Lancia, M. Pinotti, and R. Rizzi, "Haplotyping populations by pure parsimony: Complexity of exact and approximation algorithms," *INFORMS Journal on computing*, vol. 16, no. 4, pp. 348–359, 2004.

[75] A. Clark, "Inference of haplotypes from PCR-amplified samples of diploid populations," *Molecular Biology and Evolution*, vol. 7, no. 2, pp. 111–122, 1990.

[76] L. Tininini, P. Bertolazzi, A. Godi, and G. Lancia, "Collhaps: A heuristic approach to haplotype inference by parsimony," *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, vol. 7, pp. 511–523, 2010.

[77] B. Halldrsson, V. Bafna, N. Edwards, R. Lippert, S. Yooseph, and S. Istrail, "A survey of computational methods for determining haplotypes," in *Computational Methods for SNPs and Haplotype Inference*, ser. Lecture Notes in Computer Science, S. Istrail, M. Waterman, and A. Clark, Eds. Springer Berlin / Heidelberg, 2004, vol. 2983, pp. 613–614.

[78] D. Gusfield, "Haplotype inference by pure parsimony," in *Combinatorial Pattern Matching*. Springer, 2003, pp. 144–155.

[79] L. Wang and Y. Xu, "Haplotype inference by maximum parsimony," *Bioinformatics*, vol. 19, no. 14, pp. 1773–1780, 2003.

[80] D. Brown and I. Harrower, "Integer programming approaches to haplotype inference by pure parsimony," *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, vol. 3, no. 2, pp. 141 –154, april-june 2006.

[81] M. Stephens, N. Smith, and P. Donnelly, "A new statistical method for haplotype reconstruction from population data," *The American Journal of Human Genetics*, vol. 68, no. 4, pp. 978–989, 2001.

[82] T. Niu, X. Lu, H. Kang, Z. Qin, and J. Liu, "Haplotype inference and its application in linkage disequilibrium mapping," in *Computational Methods for SNPs and Haplotype Inference*, S. Istrail, M. Waterman, and A. Clark, Eds. Springer, 2004, pp. 48–61.

[83] E. Halperin and E. Eskin, "Haplotype reconstruction from genotype data using imperfect phylogeny," *Bioinformatics*, vol. 20, no. 12, pp. 1842–1849, 2004. [Online]. Available: http://bioinformatics.oxfordjournals.org/content/20/12/1842.abstract

[84] J. Liu, "The collapsed Gibbs sampler in Bayesian computations with applications to a gene regulation problem." *Journal of the American Statistical Association*, vol. 89, no. 427, pp. 958–966, 1994.

[85] R. Chen and J. Liu, "Predictive updating methods with application to Bayesian classification," *Journal of the Royal Statistical Society. Series B (Methodological)*, vol. 58, no. 2, pp. 397–415, 1996.

[86] T. Niu, Z. Qin, X. Xu, and J. Liu, "Bayesian haplotype inference for multiple linked single-nucleotide polymorphisms," *The American Journal of Human Genetics*, vol. 70, no. 1, pp. 157–169, 2002.

[87] L. Excoffier and M. Slatkin, "Maximum-likelihood estimation of molecular haplotype frequencies in a diploid population," *Molecular Biology and Evolution*, vol. 12, no. 5, pp. 921–927, 1995.

[88] Z. Qin, T. Niu, and J. Liu, "Partition-ligation–expectation-maximization algorithm for haplotype inference with single-nucleotide polymorphisms," *The American Journal of Human Genetics*, vol. 71, no. 5, pp. 1242–1247, 2002.

[89] G. Kimmel and R. Shamir, "GERBIL: Genotype resolution and block identification using likelihood," *Proceedings of the National Academy of Sciences*, vol. 102, no. 1, pp. 158–162, 2005.

[90] P. Scheet and M. Stephens, "A fast and flexible statistical model for large-scale population genotype data: applications to inferring missing genotypes and haplotypic phase," *The American Journal of Human Genetics*, vol. 78, no. 4, pp. 629–644, 2006.

[91] G. Celeux, M. Hurn, and C. Robert, "Computational and inferential difficulties with mixture posterior distributions," *Journal of the American Statistical Association*, vol. 95, no. 451, pp. 957–970, 2000.

[92] K. Liang and X. Wang, "A deterministic sequential Monte Carlo method for haplotype inference," *IEEE J. Sel. Topics Signal Process.*, vol. 2, no. 3, pp. 322–331, 2008.

[93] C. Tsallis, "Possible generalization of Boltzmann-Gibbs statistics," *Journal of Statistical Physics*, vol. 52, no. 1, pp. 479–487, 1988.

[94] A. Marshall and I. Olkin, *Inequalities: Theory of Majorization and Its Applications*. Academic Press New York, 1979.

[95] A. Martins, P. Aguiarz, and M. Figueiredo, "Tsallis kernels on measures," in *Proc. IEEE 2008 Information Theory Workshop*, May 2008, pp. 298–302.

[96] C. Papadimitriou and K. Steiglitz, *Combinatorial Optimization: Algorithms and Complexity*. Dover Publications, 1998.

[97] A. Phillips, J. Rosen, and M. Vliet, "A parallel stochastic method for solving linearly constrained concave global minimization problems," *Journal of Global Optimization*, vol. 2, no. 3, pp. 243–258, 1992.

[98] G. Cornuéjols, "Valid inequalities for mixed integer linear programs," *Mathematical Programming*, vol. 112, no. 1, pp. 3–44, 2008.

[99] A. Krause and V. Cevher, "Submodular dictionary selection for sparse representation," in *Proc. 27th International Conference on Machine Learning (ICML 2010)*, Haifa, Israel, June 2010.

[100] G. Nemhauser and L. Wolsey, "An analysis of approximations for maximizing submodular set functionsI," *Mathematical Programming*, vol. 14, pp. 265–294, 1978.

[101] A. Krause, A. Singh, and C. Guestrin, "Near-optimal sensor placements in Gaussian processes: Theory, efficient algorithms and empirical studies," *The Journal of Machine Learning Research*, vol. 9, pp. 235–284, 2008.

[102] D. Hartl and A. Clark, *Principles of Population Genetics*. Sunderland, Massachusetts: Sinauer Associates, Inc., 1997, vol. 3.

[103] M. Rieder, S. Taylor, A. Clark, and D. Nickerson, "Sequence variation in the human angiotensin converting enzyme," *Nature Genetics*, vol. 22, no. 1, pp. 59–62, 1999.

[104] B. Kerem, J. Rommens, J. Buchanan, D. Markiewicz, T. Cox, A. Chakravarti, M. Buchwald, and L. Tsui, "Identification of the cystic fibrosis gene: genetic analysis," *Science*, vol. 245, no. 4922, pp. 1073–1080, 1989.