ALGORITHMS FOR RATIONAL VACCINE DESIGN

by

Vladimir Jojic

A thesis submitted in conformity with the requirements
for the degree of Doctor of Philosophy
Graduate Department of Computer Science
University of Toronto

Abstract

Algorithms for rational vaccine design

Vladimir Jojic

Doctor of Philosophy

Graduate Department of Computer Science

University of Toronto

2007

Design of an HIV vaccine has proven to be a difficult challenge. Vaccines designed by the traditional approach based on using a single weakened virus or a portion of a virus have not provided sufficient protection. The large variability of the HIV virus population is believed to be the main cause of failure for the vaccine candidates. In this thesis, I introduce an immunologically motivated vaccine score which accounts for the variability of the target HIV population. The exact optimization of this score is an NP-hard problem. I introduce algorithms, both approximate and exact, for designing a high scoring vaccine. The approximate methods are based on expectation maximization methods and use approximate probabilistic inference. The exact method is based on a branch-and-cut method for the asymmetric orienteering problem. The score and the algorithms for maximizing the score are validated both in-silico and in-vitro. In-silico comparisons are made to other methods aimed at overcoming HIV variability. The in-vitro Elispot experiments demonstrate an expected protection in $\sim 90\%$ of infections.

Acknowledgements

My gratitude goes out to my advisor Brendan Frey. He has provided me with intellectual, financial and editorial support before and during the writing of this thesis. His guidance throughout my graduate studies has been invaluable to me. I am also grateful to my committee members, Quaid Morris, Ryan Lilien, Radford Neal, as well as my external examiner Simon Mallal, whose comments made significant impact on the quality of this thesis.

Thank you to all the members of the PSI group for providing a lively and stimulating atmosphere.

Interaction with Microsoft Research's Machine Learning and Applied Statistics group has shaped both this thesis and my understanding of the machine learning area. In particular, I have been lucky to work together with David Heckerman, Chris Meek and my brother Nebojsa Jojic. Nebojsa's help has been invaluable in many respects, scientific and otherwise.

I am thankful to the University of Toronto and Microsoft Research for the funding which has made completing my graduate studies that much easier.

Contents

Chapter 1

Introduction

This thesis deals with the design of a synthetic preventative vaccine for the human immunodeficiency virus (HIV). I formulate the problem of vaccine design as an optimization problem. The essential part of this problem formulation is a vaccine score based on the biology of HIV and human immune system, specifically T-cell response. I also propose methods, some approximate and some exact, for producing vaccines with a high score. The biological experiments indicate higher levels of protection by vaccines designed using my methods than the vaccines produced by the previously proposed methods.

Previous attempts at HIV vaccine design have failed in providing the desired level of protection against the infection. One of the challenges facing HIV vaccine development lies in the enormous variability of the HIV virus [30, 9, 75, 34, 61], which stems from the ability of the virus to mutate rapidly. The diversity of human immune types further increases the challenge of designing a single universal vaccine, as the different immune system types detect different signals of infection. I propose vaccine design methodology that addresses both of these sources of variability.

In this thesis, I focus on the methodology of developing a T-cell vaccine against HIV. A vital way in which T-cells contribute to the human immune response is by destruction of infected cells. The T-cells recognize infected cells by detecting short viral peptides exhibited on the

surface of the cell. These viral peptides are short contiguous fragments of the viral protein of length of 8-10 amino acids. The viral peptides can originate from any position in a viral protein; hence the number of viral peptides from a single viral protein is only slightly smaller than the length of viral protein. Since the virus mutates, many variants of the protein exist and the set of viral peptides is quite large. There are two further characteristics of the viral peptide that play a role in the methods developed in the thesis. First, not all viral peptides can be recognized by all of the immune system types. The set of viral peptides provided by the vaccine should contain suitable peptides for each of the immune system types. Second, there is a significant amount of overlap between peptides, e.g., the amino acids at the end of one peptide are the beginning amino acids of another peptide.

The immune response is greatly improved if the viral peptides have been "learned" by the immune system prior to the infection. It is the role of a vaccine to provide examples of viral peptides for the immune system to learn. The learning mechanism is essentially the same mechanism used for detecting the presence of a virus in a cell. The fragments of the vaccine are presented on the cell wall and detected by the immune system. It seems natural, then, that creating vaccines containing a large number of viral peptides would provide good protection. However, compressing the large set of peptides required into a compact vaccine, which in itself is a string of amino acids, seems to be a quite difficult task. The key observation is that many of the peptides have significant overlap, since they stem from neighboring positions in the viral protein.

Previously proposed algorithms for designing synthetic vaccines [33, 64] were based on the assumption that the diversity challenge can be overcome by utilizing a single "central" sequence similar to all of the variants of the virus. This similarity is defined in terms of position-wise distance between sequences: evolutionary distance was used in the case of the Center of Tree (COT) and related ancestral methods [64]; a distance based on the number of mismatches between sequences was used in case of consensus methods [66]. The underlying assumption is that the "centrality" of the sequence entails recognition of viral peptides in the viral proteins

by an immune system trained on viral peptides in the vaccine . Due to the high variability of the virus, this assumption need not hold outside of highly conserved regions.

In this thesis, I propose a method founded on a vaccine score that takes into account the nature of T-cell immune response and its basis in detection of viral peptides. The score of a vaccine is the fraction of all viral peptides present in the vaccine. The set of viral peptides is obtained from a dataset of viral sequences; in this thesis I use the set of sequences referred to as the Perth dataset from [60]. The set of viral peptides may be restricted to peptides from a particular viral region or viral protein. Hence, peptides that occur in many viral sequences, the more frequent peptides, will contribute more to the coverage score of a vaccine than the less frequent ones. The larger the score, the more likely it is that a viral peptide occurred in both the vaccine and an infecting virus. If the peptide had been learned by the immune system of a vaccine recipient, then the infection can be efficiently cleared by the recipient. In this thesis, I assume that frequency of appearance of peptides in the vaccine translates to increased learning of the peptide by the immune system.

I propose algorithms for finding a vaccine with a high score. I show that the problem of finding the optimal vaccine is NP-hard, necessitating use of approximate algorithms to solve the problem. The essence of the algorithms lies in exploiting the overlap in the viral peptides in order to produce a compact vaccine. The algorithms include a greedy algorithm, closely related to sequence assembly algorithms [27], and a number of algorithms based on probabilistic inference, inspired by the epitome model [39], and an exact algorithm, based on the branch-and-bound ideas. Other challenges facing vaccine design such as leveraging cross-reactivity, utilizing multiple coding frames, avoiding auto-immune response and immunodominance are addressed at the end of the thesis and point towards possible extensions of the methodology.

The thesis starts off with a review of relevant immunology: in particular, the basics of the formation of the CD8 T-cell repertoire, as well as the role of the human leukocyte antigen types. A review of the HIV life cycle is provided next. The last portion of the review focuses on previously-proposed vaccine design methods and the state of the art in vaccine design. A num-

ber of predictive methods which may be used to guide the vaccine design are also presented. Readers familiar with the material of these review sections may wish to skip them and proceed to chapter 4 which introduces the contributions I have made: the vaccine score, algorithms for optimization of the score and vaccine design comparisons. Section 5.2 contains experimental results comparing the vaccine designed using the proposed algorithms and a consensus vaccine in ELISPOT experiments, which test whether peptides are indeed epitopes for a population of patients. Chapter 6 discusses modifications to the methods aimed at overcoming some of the additional challenges to vaccine design.

Chapter 2

Immune system and HIV infection

Successful vaccines against diseases such as mumps, measles, and flu, utilize a single live or inactive strain of the pathogen. A single strain is sufficient to stimulate the immune system to learn to respond to the pathogen quickly. HIV/AIDS poses a challenge to vaccine design stemming from the enormous variability among HIV strains; the single strain approach has so far failed to produce an efficacious vaccine. I will propose an approach to designing synthetic vaccines which stimulate an immune response to a wide range of different HIV strains. This approach is based on training the killer T-cells to recognize infected cells based on short peptides, epitopes, on the cell wall. The goal of vaccination is to expose appropriate T-cells to epitopes which may be encountered in the real infection. In the following section, I will outline the immunology involved: the mechanisms which lead to exposing the short peptides on the cell wall, "processing and presentation," and the process of selection that leads to a population of T-cells capable of disambiguating between infected and non-infected cells. The next sections deal with the life cycle of HIV. The last section deals with vaccine design.

2.1 The immune system

Two arms of the immune system– the nonspecific (or innate) arm, and the specific (or adaptive) arm – protect an organism against invading pathogens. Their goal is to remove all pathogens

from of the body, clearing the infection.

The innate arm detects structural features shared by many pathogens. For example, the nonspecific complement system, which consists of a number of different proteins, targets the surface of cells with the aim of puncturing the cell wall and killing the cell. The organism's own cells have surface molecules that turn off the complement attack. In addition, the innate arm also employs physical and chemical barriers, and reduces access to essential nutrients like iron. However, the mechanisms employed by the innate arm, and the range of pathogens it can target, do not change over the life of the organism.

The adaptive arm is constantly learning from exposure to different pathogens, and hence it is changing throughout the life of the organism. Through vaccination, the adaptive arm offers us the ability to train the immune system to respond to infections *before* they occur. In the following section, I present a simplified overview of a viral infection cycle and how an already trained immune system responds. Following sections address details of training and response.

2.2 Viral infection and response

The adaptive immune system can learn to respond to a wide variety of infecting organisms. Here, I focus on the viral infection cycle and the immune system response to this particular type of infection. Viruses consist of genetic material in a capsid, a protein shell. The virus cannot reproduce on its own as it lacks the ability to build proteins and to replicate genetic material. In order to reproduce, the virus needs to enter a cell. The virus achieves this by attaching to the cell wall and fusing with the cell. The viral genetic material is then delivered into the cell and the infected cell starts producing the encoded viral proteins. The viral proteins are assembled into a new capsid into which copies of genetic material are marshalled. The new viruses are released either by destruction of the cell, or by having viruses bud onto the cell wall, thus avoiding destruction of the cell.

Note that the viral proteins are produced side-by-side with the proteins of the organism itself, the self-proteins. Inspection of the proteins produced allows detection of cells that are producing foreign protein. This is exactly what the adaptive immune system attempts to do in order to detect the infected cells. Proteins in a cell are regularly broken down by proteasomes into shorter peptides, a process known as proteolysis. Some of these short peptides are transported by major histocompatibility molecules (MHC) molecules to the surface of the cell. T-cells have a propensity to bind to MHC molecules. A particular type of T-cell, the cytotoxic lymphocyte or CTL, will kill cells that carry MHCs with foreign peptides. The pathway which produces a short peptide and transports it to the cell wall is called the "processing and presentation" pathway. The next section treats in detail different steps in this pathway.

2.3 How the cell signals its infection

"Processing and presentation" refers to the generation of a signal on the cell wall that a foreign peptide is being created. This signal is physically realized as a complex composed of a MHC molecule and a short foreign peptide. I will denote such a complex as MHC:peptide.

This first part of the mechanism, "processing," consists of breaking down the protein into short peptides and the subsequent trimming of these peptides. The proteasomes are responsible for producing short peptides by cleaving the protein. The cell can express two different types of proteasomes: the constitutive proteasomes and immunoproteasomes. The constitutive proteasomes are always expressed and are necessary for normal operation of the cell. The immunoproteasomes are activated by immune cytokines which signal a need for immune response. The proteasome cleavage is a stochastic process and cleavage products come from overlapping fragments [68]. This allows for a thorough sampling of peptides from the protein. The immunoproteasome and the constitutive proteasome have somewhat differing cleavage motifs and produce overlapping but distinct sets of peptides [22]. These peptides range from 3 to 22 amino acids. Less than 20% are 8-10 amino acids long. The peptides shorter than 8 amino

acids are not used in the subsequent steps of processing and presentation. The distribution of lengths is well approximated by a log-normal distribution [45]. Some of the resulting peptides are extended on the N-terminus and require further trimming by amino-peptidases [11]. The trimming is necessary in order to produce a peptide that can be bound by an MHC molecule. Two types of MHC molecules are involved in binding peptides: MHC-I and MHC-II. MHC-I is present in most cells; it binds peptides which are produced by the cell and are 8-10 amino acids long. A short peptide, 8-10 amino acids long, is transported by the Transporter associated with antigen processing (TAP) molecule to the endoplasmic reticulum, where an MHC-I molecule binds to the peptide and transports it to the surface of the cell. Peptides bind to MHC-I molecules by landing in a groove formed by two alpha helices and supported by a beta sheet. Only two or three of the amino-acids on the peptide are anchored onto pockets in the MHC-I groove and these amino acids are called "anchors". The 9mer peptides may exhibit a kink to allow proper alignment of the anchors to the corresponding pockets in the groove. The MHC-I and the bound peptide are transported to the cell wall. When the groove and the bound peptide are positioned extracellularly, we say that the cell is presenting a peptide. Figure 2.1a shows an example of a peptide bound to MHC-I molecule.

MHC-II is present on a few types of cell responsible for uptake and processing of pathogens, such as macrophages, dendritic cells and B cells; it presents peptides that are imported into cell from the surrounding environment and range from 13 to 25 amino acids in length. The MHC-II molecules in endoplasmic reticulum bind a molecule called the invariant chain. The invariant chain blocks binding of MHC-II to peptides in endoplasmic reticulum. The MHC-II becomes available for binding to peptides only after it has left the endoplasmic reticulum and entered a vesicle. In a vesicle MHC-II may encounter peptides of extracellular origin. This thesis focuses on the immune response based on presentation by the MHC-I molecules. However since the peptides presented by MHC-II are linear like the ones presented by MHC-I molecules, the main difference between peptides presented by the two types of MHC molecules is in their length. Algorithms introduced later in the thesis can be used to build vaccines which stimu-

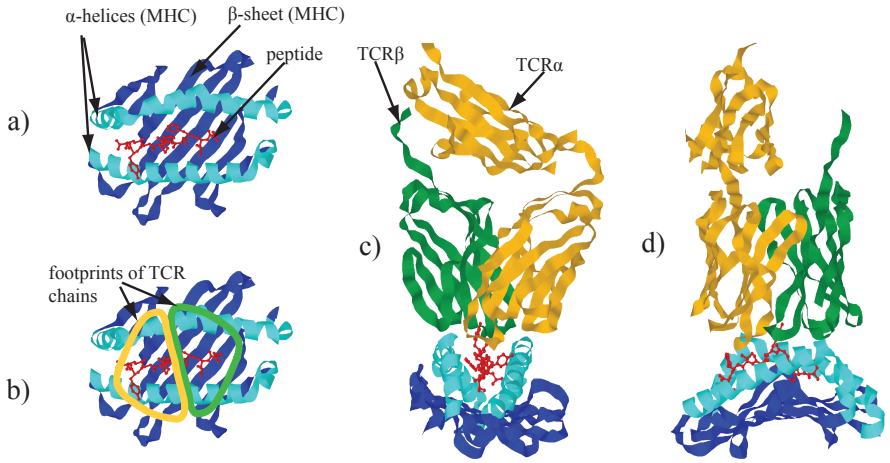


Figure 2.1: An example of MHC and TCR (T-cell receptor) interaction [32]. a) A peptide docked in a MHC groove b) The same MHC:peptide complex with outlines of the footprints of the two TCR chains c) and d) MHC, peptide and the two TCR chains from two perspectives.

late the immune system via presentation of peptides by MHC-II molecules by simply choosing peptides of the appropriate length.

Crucially, the human population as a whole has a wide range of different MHC-I types which have preferences for different, but sometimes overlapping, sets of peptides. The group of genes encoding MHC molecules is called the human leukocyte antigen (HLA) complex. HLA class I and class II genes encode particular MHC-I and II molecules and these genes jointly determine a person's HLA type. There are 3 different major HLA class genes encoding MHC-I molecules: HLA-A, HLA-B and HLA-C. These genes are all located on chromosome 6. The two copies of the chromosome can contain different copies of each of the genes, yielding a total of at least 3, and at most 6, distinct MHC-I molecules. Currently we know of 342 HLA-A, 627 HLA-B and 182 HLA-C types [54], and the list is continually revised. While the HLA types of a particular individual type may or may not confer protection from a disease, the diversity of HLA types confers protection to the population as a whole.

2.4 How infected cells are recognized and killed

A Cytotoxic T-Lymphocyte (CTL) encountering a cell presenting a foreign peptide will induce death of the cell by puncturing the cell wall. However, a single CTL can recognize only a small fraction of all possible foreign peptides; the peptides recognizable by a single CTL are said to cross-react. Equally importantly, CTLs do not induce death of the cells presenting a self-peptide, a peptide derived from the organism's own proteins. The process of producing functional T-cells which do not recognize self-peptides is called maturation. The process of stimulating a naive T-cell into production of clones is called activation. Some of these clones become effector T-cells, others memory T-cells.

2.4.1 How a diverse T-cell repertoire is shaped

In order to be able to cover a large diversity of pathogens, including viruses which will evolve in the future, T-cells themselves have to be diverse. This diversity stems mainly from the T-cell receptors (TCR) which recognize the foreign peptides, Figure 2.1b),c) and d) show the interaction between a TCR and a MHC molecule carrying a peptide. After being produced in the thymus, a T-cell attempts to produce a functional receptor through a series of gene rearrangements of V, D and J gene segments. Two distinct classes of receptors can be formed; they are designated $\alpha\beta$ or $\gamma\delta$. The differences stem from the distinct sets of gene segments. Each of the $\alpha, \beta, \gamma, \delta$ chains are formed from a separate set of V, J and, with exception of γ, D segments. In addition to the standard receptors, the T-cells express co-receptors CD4 and CD8.

The $\gamma\delta$ or $\alpha\beta$ receptors have a bias towards interacting with MHC-I and MHC-II molecules respectively, due to the structure of the V segments. However, a receptor may not bind to an MHC molecule of a particular HLA type. The first step in refining the repertoire of T-cells is the positive selection toward interaction with the organism's MHC molecules. The selection occurs in the thymus, where T-cells are exposed to MHC molecules carrying self-peptides. If a T-cell binds to an MHC molecule it receives a signal to proceed with maturation. In the absence

of this signal, the T-cell will die from apoptosis within 3-4 days after the first T-cell receptor is expressed. The α chain rearrangements continue until the death of the T-cell or successful binding. Thus, the T-cell actively explores the choices of possible receptors in pursuit of the one that binds to an MHC molecule.

The binding of a T-cell receptor to an MHC-I molecule is assisted by the CD8 co-receptor, while CD4 assists binding to MHC-II molecules. Upon binding to MHC-I, a T-cell will suppress further expression of CD4 and will only express the CD8. Similarly, upon binding to MHC-II, the T-cell will only express CD4. From this point in T-cell development, the cells can be labelled as CD8 T-cells (CTLs) and CD4 T-cells (helper T-cells) since they will express only one of the two receptors.

At the end of the positive selection, the remaining T-cells have a functional receptor, TCR, and a corresponding co-receptor, CD4 or CD8. These T-cells may also have a propensity to bind strongly to MHC molecules carrying self-peptides. This is in direct opposition to the aim of building detectors of foreign peptides; therefore, the cell undergoes an additional stage of negative selection, which rectifies this by eliminating T-cells that bind too strongly to an MHC molecule presenting a self-peptide. A T-cell which survives the two rounds of selection is called a naive T-cell, since it has not yet encountered an antigen, a real foreign peptide.

2.4.2 How a T-cell becomes a killer T-cell

The task of stimulating T-cells is performed by antigen presenting cells (APCs). Three types of APCs participate in activating T-cells in the lymph nodes: dendritic cells, macrophages and B-cells. The macrophages and B-cells are responsible for stimulating CD4 T-cells only. The dendritic cells stimulate both CD4 and CD8 T-cells. APCs acquire various macromolecules by endocytosis from the surrounding environment. The material ingested by the APCs includes viruses, bacteria and dead, possibly infected cells. A virus or bacterium may be ingested by the APC, but also dead and possibly infected cells are ingested. The ingested material is degraded and peptides are generated which undergo presentation.

Aside from the MHC peptide on the surface of APC binding to a TCR, additional co-stimulation is required to activate a T-cell. This co-stimulation is provided only by APCs, in the form of the B7 receptor which binds to the CD28 receptor on the T-cell. If a T-cell does not receive the co-stimulating signal it becomes non-responsive, effectively neutralizing the T-cell, an event called anergy. A particular naive T-cell needs to bind to an APC and receive both signals more than 100 times before it is activated.

An activated effector CD8 T-cell is capable of killing an infected cell by secreting cytotoxins or signaling a target cell to undergo apoptosis. Activated CD4 T-cells become helper T-cells which stimulate B-cells, and may help activate CD8 T-cells. An activated T-cell will proceed to secrete interleukin which in turn drives further division of the T-cell.

A memory T-cell, unlike an effector cell, remains in the thymus for long periods of time and can be reactivated by an encounter with the same antigen. The reactivation is a much faster process than the initial activation allowing the immune system to mount a fast secondary response. Vaccines aim to stimulate production of memory T-cells which can be swiftly reactivated, clearing the infection early on. The peptides recognized by activated T-cells are called epitopes. Hence the real payload of a vaccine is the set of epitopes.

2.5 HIV structure and infection cycle

HIV consists of a cone shaped capsid. The capsid is formed out of approximately 2000 copies of capsid protein and 2000 copies of matrix protein, which are stabilized by nucleocapsid protein. The capsid contains two copies of the viral genome in RNA form, three enzymes (protease, reverse transcriptase, and integrase), and three additional accessory proteins (Nef, Vif and Vpr). Three HIV proteins, Rev, Tat and Vpu, are synthesized only in the cell and are never integrated into the virus itself. Of course, these proteins are always encoded in the HIV genome.

The capsid itself is surrounded by a bilayer lipid, which contains HIV's surface protein and

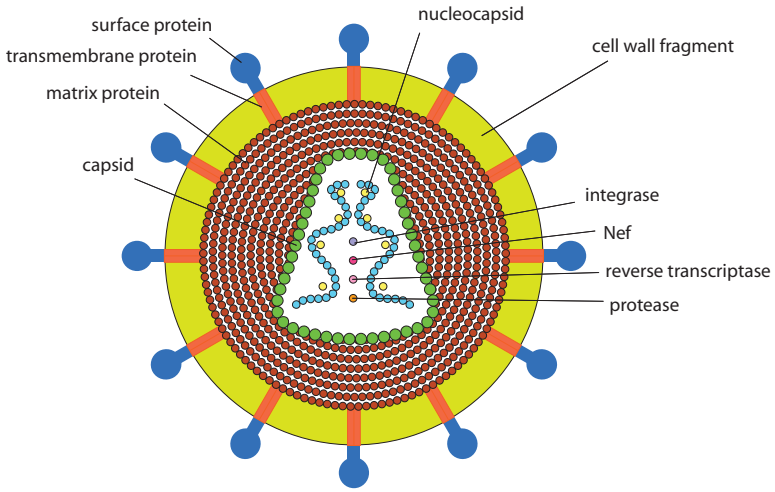


Figure 2.2: Sketch of mature HIV structure.

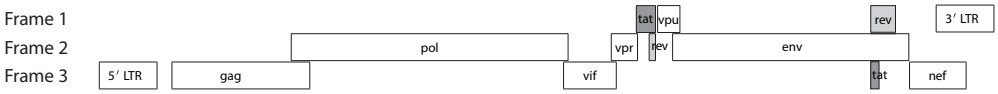


Figure 2.3: Map of HIV genome across three coding frames.

is anchored to the virus via transmembrane protein. The lipid bilayer is derived from the cell wall and may contain MHC, actin and ubiquitin.

2.5.1 HIV early phase

Generally an HIV infection begins when dendritic cells attach to the virus, or in some cases may become infected themselves. These dendritic cells travel to lymph nodes to present the captured virus, allowing HIV to come into contact with CD4 T-cells, the main target of HIV. The virus attaches to a CD4 T-cell by using the CD4 receptor and chemokine receptor CCR5, or less frequently CXCR4, and fuses with the cell.

Upon fusion, the contents of the capsid are injected into the cell. The HIV reverse transcriptase catalyzes the reverse transcription of the viral genome into DNA. The synthesized DNA is transported by an HIV preintegration complex which contains integrase, matrix pro-

tein, reverse transcriptase and Vpr proteins. Nuclear localization of the preintegration complex is directed by Vpr. After DNA arrives in the nucleus it is integrated into the host genome by catalytic activity of integrase.

2.5.2 HIV late phase

An infected cell initially produces short, spliced copies of mRNA, as it is unable to produce a fully elongated copy of the viral genome. The first products are copies of regulatory proteins Tat, Rev and Nef. Tat is a transcriptional activator which stimulates transcription elongation, causing the cell to produce viral-genome-length copies of mRNA. The full length, unspliced mRNA is exported with help of Rev outside of the nucleus.

Env, Gag and Gag-Pol are synthesized from the unspliced mRNA, the Gag-Pol less frequently due to frameshift. The Env polyprotein is postranscriptionally modified and cleaved to produce surface and transmembrane proteins. These two products of Env are assembled into viral capsids that will contain copies of the Gag-Pol polyprotein and of the HIV genome.

The early phase proteins such as Nef may present a better target, since their expression begins early in the HIV cycle. On the other hand, Env, Gag and Pol, though are more variable, are expressed later in the life cycle, seem to contain more epitopes [47].

Chapter 3

Background on vaccine design

The main challenge in producing a safe and effective HIV vaccine lies in the diversity of the viral population, both in a particular host and in the human population as a whole. Vaccines based on a single env gene, such as VaxGen's candidate, which reached Phase III trial, have so far failed to elicit responses to a wide variety of HIV viruses [10, 13]. HIV's noisy reverse transcription allows the virus to build up a diverse viral population. The estimated rate of mutation is roughly a single nucleotide out of 10000, per replication cycle [12]. In light of these observations, viral population diversity must be taken into account when choosing a vaccine strain or constructing a synthetic vaccine. Ideally, almost all of the HIV viruses should be detectable by the immune system after immunization by the vaccine. Our goal is to produce a vaccine that offers protection against as large as possible a fraction of the HIV population. This in turn means that epitopes that occur frequently in the HIV population should take precedence over rare epitopes.

The second challenge stems from the variability of human immune systems. There are approximately 1200 different HLA alleles across A, B and C loci (see Section 2.3). A vaccine containing a set of frequent epitopes presented by a particular HLA type may offer solid protection for all people with a particular HLA type. Assuming that a vaccine can be designed for each HLA type in isolation, approximately 1200 different vaccines would be required to

offer most of the human population protection against HIV. At this time, designing vaccines tailored for a particular HLA type is impractical, mostly due to the difficulty of determining frequent epitopes for each of the HLA types. However, an HIV vaccine should still aim to provide protection comparable to the ideal scenario outlined above. Hence, each of the HLA types should be able to find a number of frequent epitopes in the vaccine for the immune system to learn. Thus, a trained immune system would have a reasonable chance of clearing an infection by most strains of HIV, since any given strain is likely to contain the frequent epitopes. Note that the number of epitopes learned from the vaccine by any given immune system need not be large, as long as a sufficient portion of the HIV viral population has at least one of these epitopes.

In contrast, the task of designing a vaccine for a non-variable virus is easy. The vaccine which includes all of the epitopes of the non-variable virus is short. It is a copy of suitably weakened virus. Hence, for a non-variable virus, there is no issue about which peptides to include in the vaccine, since the set of peptides is small and they are all frequent. In this case, a particular immune system will not be limited by the fact that the epitopes it can learn are infrequent. In effect, all of the HLA types are treated equally, and the only limitation for an immune system lies in its ability to detect the virus, not in the shortcomings of the vaccine. In the case of HIV and a vaccine based on the single virus, the set of peptides in the vaccine is a fraction of all possible viral peptides, and, more importantly, peptides are not guaranteed to be frequent.

Hence, the goal of HIV vaccine design is to produce a vaccine that, for almost any HLA type, provides a chance of clearing infection by almost any virus.

3.1 HIV variability

The Env protein products are the only HIV characteristic proteins available on the surface of the virus. Across the entire HIV population, Env proteins differ in up to 30% of their amino

acids [15]. As B-cell response to a virus is predicated on the ability of antibodies to recognize the virus in an extracellular environment, the tremendous diversity of Env is the main challenge to the effectiveness of a B-cell vaccine.

Other genes exhibit a lesser degree of variability than Env (see Figure 3.1). Products of these genes cannot be detected by inspecting an HIV virus because the proteins are either isolated inside the capsid or the proteins are only present in an infected cell. However, they can be detected in an infected cell by cooperation of MHC and T-cells. In addition, a T-cell vaccine can also target Env protein since it is also synthesized in the cell before being assembled into virus. Hence a T-cell vaccine has wider range of potential targets than a B-cell vaccine.

The main source of HIV variability is erroneous reverse transcription by reverse transcriptase. The selective pressures placed upon this highly variable population are speed of replication , immune system response and drugs. These pressures shape the population within a host. The similarities in the immune systems in a particular human population, more precisely in its HLA type distribution, leave an imprint on the viruses infecting the population.

Analysis of the variability of the HIV population based on phylogenetic tree reconstruction of HIV and related viruses has led to the subdivision of the HIV population into groups. The main group (M for “main”) contains the bulk of HIV viral sequences collected worldwide. The other group (O for “outgroup”) is rare and has been found in a limited number of locations. The M group has been further subdivided into clades A-K. The clades are prevalent in particular geographical regions; a list of the 9 most frequent clades along with geographic location is given in Table 3.1.

3.2 HIV vaccine delivery, targets and constraints

The purpose of a vaccine is to stimulate the development of memory T-cells sensitive to viral epitopes. In order to achieve this, viral peptides in the vaccine must be loaded onto MHC-I molecules and presented to effector T-cells. There are two types of approaches available for

Clade	Region where the clade is dominant
A	East Africa, West Africa, Central Africa Eastern Europe, Central Asia
B	North America, South America, East Africa, Central Africa, North Africa, Middle East, Europe, Australia, New Zealand, Japan, China, Korea, Philippines, Malay Peninsula
C	India, Brazil, South Africa, East Africa, Nepal, China
D	East Africa, Central Africa, West Africa, Eastern Europe, Central Asia
F	Central Africa, West Africa, Latin America, Caribbean, North America
G	West Africa, Central Africa, North Africa, Middle East, Eastern Europe, Taiwan, Korea
H	Central Africa, Eastern Europe, Central Asia
J	Central Africa, West Africa
K	Cameroon, DR Congo

Table 3.1: The geographical distribution of HIV subtypes in the main group [58].

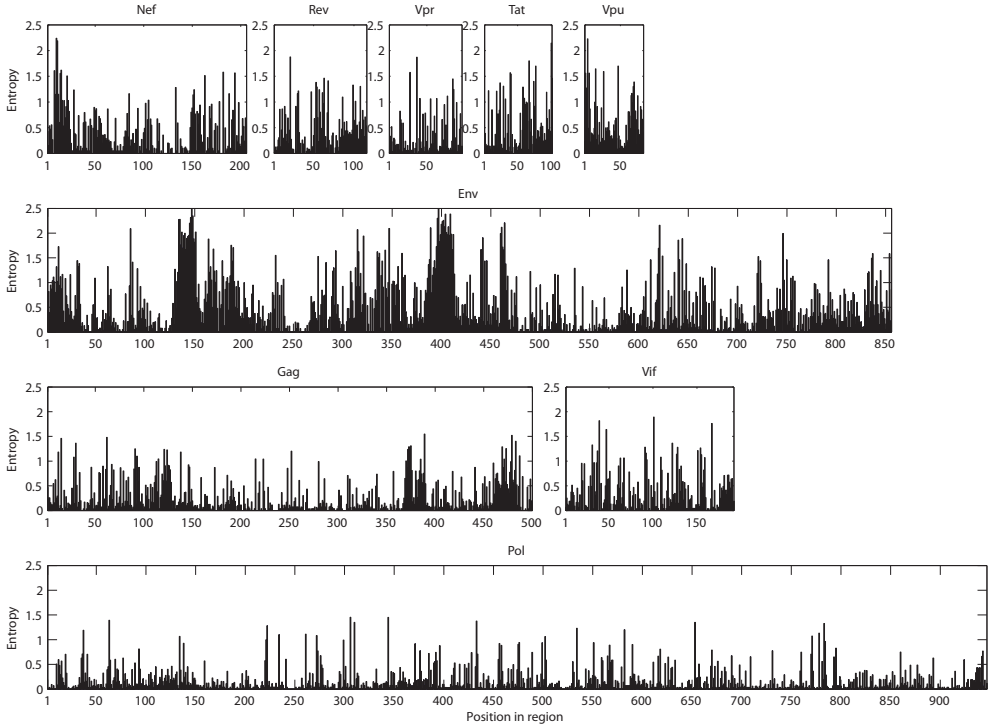


Figure 3.1: Entropy (in nats) [14] in HIV genome regions. For a given region at each offset I compute the entropy of the amino acid distribution in the HIV population for clade B. Sequences taken from [60]. All the sequences are in the amino acid domain and were aligned to a reference sequence HXB2 [46].

delivering a vaccine. The first is to deliver viral peptides to an antigen presenting cell that will process and present the foreign peptide. These viral peptides, which may be a single epitope or a longer multi-epitope peptide, are delivered in the form of a peptide bound to a liposome [19]. The second approach is to deliver genetic material to the antigen presenting cell, allowing production of viral peptides to occur inside the cell.

Further subdivision of the second approach is based on the means of delivering the genetic material and the manner in which the foreign peptide is produced. A viral peptide may be encoded in plasmid DNA which is injected into the muscle tissue and may enter an APC [80]. Other vectors for delivering genetic material into the cell are viruses: pox, adeno, alpha viruses, measles and poliovirus. Poxviruses can carry large foreign peptides (at least 25Kb [76]). Poxviruses replicate in cytoplasm and have virtually no risk of integration. Adeno viruses can carry up to 11Kb and are effective at entering a variety of cells, including antigen presenting cells. The replication competent variants of adenoviruses have smaller capacity: up to 4Kb [69]. Alpha viruses are highly immunogenic vectors due to their ability to suppress translation of host mRNA and induce high levels of viral protein expression [50]. Measles virus can infect macrophages and dendritic cells. Given that most of the adult human population has immunity to measles, this vector can only be used in children, and would provide immunity against both HIV and measles [78]. Similarly, almost the entire human population has immunity to polio viruses; thus the use of this vector is restricted to infancy. In addition, the poliovirus has a relatively small capacity of 400 amino acids [5]. An in-depth comparison of trade-offs between different viral vectors can be found in [8].

In addition to the choice of a particular vector, the choice of adjuvants, molecules meant to further stimulate immune response, influences the immunogenicity of a vaccine and the immunization schedule. For example, a combination of delivery mechanisms can be used in an immunization schedule meant to prime and boost the immune response, for example priming with DNA and boosting with viral protein [52].

The database of worldwide vaccine trials [38] illustrates the range of vectors utilized in

current HIV vaccine candidates. Most commonly, candidates utilize DNA plasmids, adeno viruses and the poxvirus modified vaccinia Ankara as vectors for delivery of genetic material. Multiple candidates utilize a prime-boost approach with different delivery vectors, for example the HVTN 204 phase II candidate. Some vaccines, in addition to delivering genetic content in a viral vector, also deliver known CTL epitopes in lipopeptide form, for example HVTN 042 candidate. Current vaccine candidates target different clades, and in some cases multiple clades; an example of this is candidate HIVIS 03 [38] which targets clades A,B,C and E. The candidates utilize a combination of different regions, most predominantly Gag, Pol, Nef and env regions.

Vaccine vector particles carrying a particular insert must be present in sufficient numbers to elicit an immune response. A vaccine which spans multiple inserts will naturally require higher doses in order to meet the immunogenicity requirements for each of the inserts. However, a large dose may result in adverse reactions to the vaccine. A case of a death due to an adverse reaction to a high dose of an adenovirus based vaccine has been reported [73]. Currently, the doses are determined by trials with escalation and monitoring for mild to moderate reactions to the vaccine, as these may indicate a threshold of well-tolerated doses.

A vaccine with epitopes such that almost any virus in the population contains some of these epitopes will have a better chance at providing protection. If such a set of epitopes is small, a short vaccine can provide adequate protection. In a variable viral population, the set of such epitopes will be large, and the issue of placing all appropriate epitopes in a 11Kbp vaccine insert becomes more challenging. The constraints of the minimum dose for immunogenicity coupled with the maximum well-tolerated dose, and the need for wide variety of epitope coverage, necessitate algorithms for creating a compact vaccine.

3.3 Synthetic vaccine design

Isolate based vaccines work only when the epitopes (see Section 2.4.2) within a particular virus, an isolate, are sufficiently similar to the epitopes in all viruses. The idea of finding a sequence similar to most HIV strains gave rise to the notion of "central sequences" [33]. This sequence may originate from a real virus, or it may be synthetic. I will call a sequence synthetic based on its origin; a sequence generated by a computational method is synthetic. This does not preclude a synthetic sequence that exactly matches a real virus. Note that the delivery mechanisms described in the previous section require only an RNA or a DNA insert; the origin of the sequence does not have any impact on building the vaccine once the insert is synthesized.

Synthetic vaccine design methods take as input a dataset of viral sequences. For simplicity, I will assume that the sequences are in the amino-acid domain.

Different choices of sequence distances lead to different notions of centrality and ultimately to different vaccine sequences. In addition, a distinction can be drawn between a sequence chosen from the dataset according to the distance and a sequence designed so as to minimize the distances to existing sequences in the dataset. I will refer to the latter as a *synthetic central* sequence. Note that a synthetic central sequence may correspond to a real virus. In the rest of this section I focus on approaches which generate such synthetic central sequences.

One simple distance between sequences is the Hamming distance [14], the number of positions at which the two sequences differ. This distance is defined only for equal-length sequences, which have been aligned. HIV virus sequences are commonly aligned to the prototypical HXB2 sequence [46]. A synthetic central sequence with respect to Hamming distance is a sequence which has the most frequent amino acid at each offset. This type of sequence is commonly referred to as a consensus sequence. Use of consensus sequences as HIV vaccines was first explored in [66].

Under Hamming distance, any two distinct amino acids are equally distant from each other. One improvement over Hamming distance is to take into account the similarity between amino acids. The substitution matrices of evolutionary models aim to capture these similarities. The

rationale is that mutation into a biochemically similar amino acid is much more likely than a mutation into a dissimilar amino acid. For example, a mutation from a hydrophobic into a hydrophillic amino-acid may change the fold of the protein.

Phylogenetic analysis of HIV populations based on such evolutionary models has shown structure in the HIV sequences found across the world and led to the formulation of clades, see Section 3.1. Phylogenetic analysis discovers an evolutionary process that could have given rise to a set of HIV sequences. This process is specified by a phylogenetic tree and a mutation model. In the phylogenetic tree, each node corresponds to a sequence (of equal length). The leaves correspond to the observed (aligned) HIV sequences in the dataset. Each of the internal nodes of the tree correspond to an unobserved ancestral sequence. Each node has one parent and two children, except the root, which has two children and no parent. The edge lengths correspond to the time required for the mutations to accumulate and separate the child sequence from the parent. The mutation model usually assumes that each position in the sequence can be independently mutated to produce a new sequence, that is, it is a site-independent mutation model. This assumption allows for efficient evaluation of the probability of sequences given a particular tree, since belief propagation, also known as a peeling algorithm [23] in the context of phylogenetic trees, can be run independently for each position in the sequence. The tree for which the evolutionary model assigns the highest probability to the observed sequences is then assumed to describe evolution of the set of sequences. Even though the assumption of the site independent mutation model makes evaluation of the probability of the observed sequences for a given tree tractable, the search for the tree that has the highest probability of producing the sequence set is still intractable and trees discovered by phylogenetic software are not necessarily optimal [24].

Given a phylogenetic tree over sequences, one of the ancestral sequences may be a reasonable vaccine candidate. One such choice is the root node corresponding to the most recent common ancestor of all the observed sequences. Depending on the structure of the tree, the most recent common ancestor may be equally distant from the observed sequences, as, for ex-

ample, in the star-like topologies of HIV-1 [3]. [1] Note that the consensus method is equivalent to the most recent common ancestor method under the following assumptions: the topology is a real star, that is to say, all of the sequences share a single parent, and each of the edges is of the same length, and probabilities of mutations between letters are uniform. On the other hand, ladder-like topologies [62], which usually best explain an intra-patient population over long periods of time, may have a most recent common ancestor which is not equidistant from all the sequences in the dataset [43]. In order to address the issue of producing central sequences with respect to evolutionary distances, the center of tree method was proposed [64]. Both ancestor and center of tree methods generate *synthetic* sequences since both are obtained by maximum likelihood estimation of a sequence corresponding to a position in the tree and need not correspond to a real ancestral sequence.

Preliminary studies of the immunogenicity of vaccines based on consensus [31] and ancestor [20] vaccines have been performed and these vaccines have not elicited larger number of immune responses than natural strain vaccines.

There are several challenges to the approaches described above. The peptides in central sequences are assumed to be able to stimulate T-cells that recognize peptides in the viral population. This assumption of equivalence between different peptides due to a T-cell's inability to distinguish between them is called cross-reactivity. As of yet, there is no analysis suggesting that the above distances, be they evolutionary or Hamming, are correlated with cross-reactivity. If such a correlation were present, the distance between two sequences would indicate whether using the first sequence as a vaccine would confer protection against the second sequence.

Moreover, both consensus and phylogeny based methods operate on sequences aligned to a prototypical sequence. The alignment of a set of HIV sequences frequently introduces "indel" symbols, corresponding to the insertion or deletion of letters in sequences. An indel artificially extends a sequence. Furthermore, if an indel is located in the middle of an epitope it may abolish the immunogenicity of the epitope by, for example, pushing anchor amino acids

[1] The star topologies are still trees but they appear like stars under a radial tree layout.

M	G	G	K	G	S	K	S	S	K	-	K	W	P	B.GB.046j
M	G	G	K	G	S	K	S	S	-	I	G	W	P	B.IT.4.IT
M	G	G	K	W	S	K	C	S	V	V	G	W	P	B.GB.044c
M	G	G	K	G	S	K	S	S	V	I	G	W	P	B.US.RP1a
M	G	G	K	G	S	K	S	S	V	I	G	W	P	(Consensus sequence)

Figure 3.2: Impact of alignment on potential epitopes. The first two sequences contain peptides GSKSSKKWP and GSKSSIGWP. The consensus sequence cannot contain these two peptides due to an indel (-) which was necessary for alignment.

away from each other. Consensus and phylogeny based sequences will not include the original epitope at this position, rather a stretched epitope with an amino acid in the place of the indel will be obtained. This situation is illustrated in Fig 3.2.

3.3.1 Cocktails of synthetic vaccines

The methods discussed in the previous section can naturally be extended to produce cocktails of vaccines. To my knowledge, the authors who proposed the original techniques have not considered this extension, but due to its simplicity, I describe this extension in the background section of the thesis. The consensus sequence is a center of the dataset under Hamming distance, since it has minimal total distance to the viral sequences in the dataset. The algorithm which produces the consensus sequence can be viewed as a K-means algorithm [53] with respect to Hamming distance, for $K = 1$. Hence, running the K-means algorithm with a larger K will produce multiple centers. After the algorithm converges, we can assign each sequence to the closest center, forming sequence clusters. The consensus sequences for each of the clusters can be used as a vaccine component. It is important to note that in the case of $K = 1$, the K-means algorithm converges to a global minimum. In the case of a larger K, we do not have a guarantee that the global minimum will be found.

Similarly, we can define a cocktail of centers of tree, where each member of the cocktail is a center of a sub-tree for a particular cluster of sequences. A single center of tree is found by determining the position in the tree that has the least total evolutionary distance from the tips of the tree, i.e., the observed viral sequences. Unlike the ancestor method, the center of the COT method is not required be among nodes of the tree and the center can be located anywhere on the edges of the tree. However, it can be shown that there exists an optimal solution for the centers of tree problem where each center is located at a node [36]. Hence, the problem of finding K centers of tree reduces to choosing K nodes in the tree with optimal distance to the leaf nodes. Furthermore, the centers of tree problem, for a given phylogenetic tree, can be solved exactly in polynomial time $O(Kn^2)$ [1] where K is number of centers and n is number of sequences.

Note that both of these algorithms, like the non-cocktail versions, use a dataset of aligned viral sequences. The alignment algorithms produce sequences of the same length. Some of the aligned sequences may contain so called "indel" characters. The resulting vaccines will be of length $k \times AL$, where k is the number of components in the cocktail, and AL is the length of a single aligned sequence.

3.3.2 Underlying assumptions about HIV and immunology

As far as I am aware, the first published work to investigate building vaccines out of short peptides is reported in [79, 40] by myself and collaborators. Hence there is no precedent on how such vaccines may perform. In fact, the in-vitro experiments in this thesis are, to my knowledge, the first assessment of such vaccines. Underlying the computational and in-vitro experiments presented in this thesis are certain assumptions about the biology of HIV and immunology. The reader should not consider these assumptions to be prerequisite for application of the methods presented in this thesis; the last part of the thesis investigates how the framework can be modified to accommodate a variety of assumptions, some not made by the author in the experiments presented here.

The first assumption deals with the set of peptides from which the vaccine should be derived. I assume that this set should include as many of the viral peptides from across the HIV population as possible, since including more examples of the viral peptides increases the probability that any given HLA type will be able to discover epitopes it can learn. It has been shown that viruses in a particular patient evolve so as to avoid detection by the host's immune system [60]. Hence a particular virus may contain only peptides which are not epitopes for the host's immune system. However, it has also been shown that a virus making a transition between two hosts with differing HLA types can reverse the escape mutations, which rendered its peptides invisible to particular HLA type, that are not necessary for survival in the new host with different HLA types[26]. Hence the set of peptides from which to build the vaccine should be derived from a diverse set of viruses, preferably obtained from a heterogeneous population of patients.

Second, I assume that an epitope that has been learned by the immune system of an HIV infected patient can also be learned from the vaccine. The actual number of epitopes that can be learned from a vaccine may be modified by such matters as immunization schedules [56], codon usage [17] and immunodominance [51]. It has been observed that the number of epitopes learned from a vaccine can vary from just 1 epitope up to 12 [56] epitopes. Except in the case of "string of beads" vaccines [16] that contain a small number of epitopes spaced out in the vaccine, it is generally not known how many epitopes are available in the vaccine for a given HLA type. This is due to the fact that most known epitopes were discovered by exposing peptides to T-cells from an infected patient. The T-cells of an infected patient need not have learned the whole set of epitopes available in the viruses, so a lack of response to a peptide does not necessarily mean that the peptide is not an epitope.

Third, I assume that a vaccine with 50% efficacy- that is, a vaccine that provides protection in 50% of the infections– is sufficient to impact the spread of HIV [4].

Considering these assumptions, we may wonder if a single isolate vaccine may provide sufficient protection. The vaccine trials so far have not demonstrated sufficient efficacy of a

single isolate vaccine. This does not mean that such a vaccine could not exist; rather, that finding such a vaccine is growing less likely. I will now examine whether the assumptions listed above imply the outcome that a single isolate vaccine is efficacious. Let us assume that we have picked a single Gag vaccine and that each patient can learn R epitopes from this vaccine regardless of their HLA type. We can use sequences from the Perth dataset [60] as possible vaccine candidates and estimate how well they would protect against infection by other viruses in the dataset, under the assumption that R is $1, 10, 25, 50 and 75$. For this experiment, we assume that the sets of learnable epitopes are randomly chosen 100 times for each value of R. We find that the average protection for each R afforded for these values of R are $0.79(0.01)\%, 7.45(0.66)\%, 14.77(1.15)\%, 41.69(3.53)\% and 51.44(4.33)\%$, respectively, with standard deviations given in the parenthesis. Hence, under an assumption of uniform distribution of epitopes across Gag, a single strain vaccine from the Perth dataset would have to contain 75 epitopes in order to offer protection to a single patient with 50% efficacy. Given the length of the vaccine and the large number of different HLA types, it is unlikely that every patient would be able to learn 75 epitopes from this vaccine. In addition, R is greater than the maximal numbers of epitopes reported in the literature, 10-12 [65, 56]. Hence, according to the assumptions made above, it is unlikely that a vaccine based on a single Gag gene would be able to offer 50% efficacy.

In order to achieve better efficacy, we can build longer vaccines in order to accommodate peptides for different HLA types, and we can also aim to lower R. The lowering of the required number of epitopes can be achieved by including more frequent epitopes in the vaccine. In the extreme and unlikely case of having an epitope which occurs in all of the viruses and is recognizable by all HLA types, efficacy of 100% could be achieved with $R = 1$. A much more likely scenario is that a small set of epitopes with moderate frequency of occurrence in the viral population may provide sufficient protection. It is crucial to note that the actual count of epitopes learned from the vaccine is not as important as their overall frequency in the viral population. The drawback of utilizing a larger number of epitopes is that the delivery

mechanisms of the vaccine may not allow the immune system to learn such a large number of peptides. Clearly any advance which allows the immune system to learn a larger number of epitopes from the vaccine would relax requirements on population frequency of the peptides in the vaccine.

The methods presented in this thesis maximize overall population frequency of peptides in the vaccine. Given a fixed capacity for the vaccine insert, the peptides which occur in large number of viral sequences, the more frequent peptides, are preferred for inclusion in the vaccine. In the extreme, we may be able to build a vaccine which includes all of the viral peptides from a particular gene. Whether this vaccine would outperform a shorter vaccine which contains smaller set of frequent peptides is unclear. Certainly all of the epitopes would be present in the longer, encyclopedic, vaccine. However, immunodominance may restrict the ability of the immune system to acquire all of the epitopes. Population frequency of the acquired epitopes may be low. In contrast, while the shorter, more selective vaccine will have more frequent epitopes, some HLA types may be underrepresented in comparison to the encyclopedic vaccine. Without an actual vaccine trial measuring T-cell responses in vaccinees, such vaccines can not be directly compared. In this thesis, a constraint on the length of the vaccine is assumed and encyclopedic vaccines are not pursued. An approach based on pruning infrequent peptides, referred to as negative design, is suggested among the extensions to the vaccine design framework. The in-vitro results suggest that a Nef vaccine designed with capacity restriction may be able to achieve sufficient efficacy even when the number of acquired epitopes per patient, R, is as low as 4. With $R = 12$, theoretical efficacy reaches above 83%, and with unbounded R, 90% theoretical efficacy can be achieved.

3.4 Prediction in vaccine design

An ideal vaccine would stimulate the creation of a number of memory T-cells that would be able to clear most of the circulating viruses. In order to establish such a library of memory T-cells,

a wide variety of presentable HIV epitopes should to be integrated into the vaccine. Hence if we knew all the epitopes in an HIV population, and understood perfectly the presentation and recognition mechanisms, we would be able to create an ideal vaccine, albeit an overly long vaccine which we might not be able to deliver in a vaccine vector.

In the absence of complete information, one can follow two approaches. The first approach is to leverage the partial information to construct explicit probabilistic models of underlying mechanisms of processing, presentation and recognition. The second approach is to be agnostic about which sequence is an epitope. One can consider each possible 8-10 amino-acid long stretch as a potential epitope. The second approach avoids potential biases in the data; for example, the bulk of the known epitopes today are associated with MHC molecules for the HLA-A region, predominantly HLA-A0201 [72], due to extensive studies of this particular HLA type.

3.4.1 Physical interaction models versus machine learning approaches

The processes underlying immunological responses involve a number of different biomolecules of significant complexity. It is possible to simulate, for example, the physical interaction between an MHC molecule and a peptide [28, 59]. Such a system consists of thousands of atoms, and run times of simulations at sufficient time-resolution quickly reach tens of hours on modern CPUs. Further, these experiments would have to be repeated for different peptides and different HLA types. A further hindrance to the simulation approach is the lack of crystallographic structures for different MHC molecules which are needed for simulation. These issues significantly complicate simulating MHC-peptide binding from first principles.

Another avenue open to us is to bypass the computationally expensive physical simulations with a simpler model. Such a model would have to represent the physics of the chemical interactions by matching the probabilities of events of interest. The supervised learning paradigm addresses the problem of training such models from experimental data. In the context of the MHC-peptide interaction, the data would consist of the sequence of a peptide and the HLA

type of the MHC molecule. Additional information about the binding of MHC and a peptide could take two forms. First, a binary label suggest whether or not the MHC and peptide bind sufficiently strongly for presentation and recognition. In this case, our task would be classification based on the MHC molecule and the peptide for which one wishes to predict binding. Second, one might use features of MHC molecules and peptides to predict measured free energy of binding in a regression task. The resulting predictor of the free energy of binding could be used as a compute binding probability between an MHC molecule and a peptide.

The following section reviews methods based on these two approaches.

3.4.2 Probabilistic models of processing and presentation

Prediction of proteasomal cleavage has been addressed by a neural network-based predictor called "netchop" [42]. Netchop is an artificial neural network trained on in-vitro data of constitutive proteasome products and MHC-I ligands. The precursors of MHC-I ligands are produced both by proteasome and immunoproteasome and are subsequently modified by aminopeptidases. Hence, training on the MHC-I ligands helps to learn the specifics of both types of proteasomes. For this method 65% of predicted cleavage sites were true cleavage sites and 15% of true cleavage sites were falsely predicted as non-cleavage sites. Another predictor, Pcleavage [7], is based on a support vector machine and for this method 86% of predicted sites were cleavage sites and 39% of true cleavage sites were falsely predicted as non-cleavage sites.

Prediction of TAP binders has been addressed by various predictors [82, 6]. Different predictors were suggested based on support vector machine, artificial neural network and a hidden Markov model. Rates achieved by the best model are: 84% peptides predicted as binders were true TAP binders and 34% of true TAP binders were falsely predicted as non-binders.

A model based on logistic regression which jointly models the HLA types and CTL recognition was proposed by [37]. Direct prediction of binding energies from a variety of data including both the measured energies and categorical data has been addressed by [41]. Au-

thors addressed the absence of crystallographic structures by using a so-called double threading model to predict the structure of the MHC molecules and peptides. Larsen et. al [49] developed a neural network predictor of the whole processing and presentation pathway, including cleavage, TAP transport and MHC binding. The method predicted as epitopes 85% of all known HIV epitopes when training on non HIV epitopes.

An interesting alternative to modeling epitope binding directly was put forward by [60]. They investigated associations between HLA types and mutations away from the consensus sequence of HIV. They were able to find mutations which would render a particular epitope invisible to an HLA type, so-called escape mutations, purely by analyzing the population of HIV viruses in patients with different HLA types.

For the purpose of epitope discovery, one may prefer to use an epitope predictor with a high true positive rate, since epitope validation experiments are expensive. However, for the purposes of vaccine design, predictions with a low false negative rate are preferable, since this regime is less likely to eliminate an epitope. For example, predictors which utilize known binding motifs may assign a low score to a peptide which does not match the motifs; however, the peptide may still strongly bind to the MHC molecules due to conformation [55]. Furthermore, the discovery of new epitopes usually starts with scanning the protein for motif matches indicating potential epitope candidates. Hence the databases of epitopes used in training predictive methods are inherently biased towards known binding motifs.

3.5 In vitro experiments for assaying peptide binding and T-cell recognition

To learn to predict peptide binding, we need data on binding affinities between peptides and MHC molecules as well as T-cell receptors. “Competition assay” is a technique to measure the binding energies of a peptide to a particular molecule in vitro. A reference peptide which is known to bind to the molecule is chosen. I will refer to the peptide of interest as the non-

reference peptide.

The reference peptide is tagged with a label. A fixed concentration of the molecule which binds both reference and non-reference peptide is introduced in a solution. The concentration of the bound, labeled reference peptide is measured. The non-reference peptide is introduced in varying concentrations so as to produce a 50% reduction of the concentration of the bound, labeled reference peptide. This concentration is IC_{50}, the inhibition concentration at 50%, for the competing non-reference peptide. Comparison of the IC_{50} for different peptides indicates stronger or weaker binders. Thresholding the IC_{50} produces categorical data. Predicting the categorical data is then a classification task (Section 3.4.1). Predicting the IC_{50} directly is a regression task. Note that measurements of the IC_{50} are repeated a number of times in order to find via curve fitting the exact IC_{50}. Hence the additional uncertainty in the measurement may also be modeled. This kind of experiment is useful for assessing binding of peptides to TAP and MHC molecules.

Measuring the T-cell detection of an antigen is usually performed by the ELISPOT. Antibodies specific to a cytokine released by an activated T-cell are immobilized on a plate. The antigen, peptide or MHC:peptide, and T-cells are deposited on the plate. If the T-cell recognizes the antigen it secretes cytokine which in turn binds to the antibodies on the plate, while unbound T-cells and antigen are washed away. Subsequently a labeled antibody will bind to the cytokine as well. The labeled molecules can be counted to indicate the number of successful activations of T-cells.

Chapter 4

Computational vaccine optimization

Ideally a line of T-cells detecting a single epitope conserved in all of the viruses would be sufficient to kill all of the infected cells. If this epitope could be presented by all of the human HLA types, a simple vaccine carrying the epitope would be sufficient to train the desired line of T-cells. In the case of HIV, such an epitope does not exist, in part because of the variability of the HIV virus, but also because different HLA types present different peptides. Which peptides should be included in the vaccine? The peptides which are shared by a large number of HIV viruses and have potential to be epitopes. The complete list of potential epitopes should include both the known epitopes (for example, those in the Los Alamos HIV epitope database [47]), the peptides occurring in HIV strains which may be deemed epitopes by prediction algorithms but are not yet verified, and the epitopes which will evolve under the selective pressure of drugs and vaccines. Note that evolution to escape the selective pressure of a drug is not site independent and it is generally focused in regions targeted by the drugs [71]. The set of potential epitopes is not well defined, since we do not know the direction of evolution of HIV under pressure from yet to be introduced drugs. To deal with the former problem we may hypothesize a model of escape mutations based on existing drugs and their imprint on the HIV sequences in patients treated with the drugs. On the other hand, in the absence of novel selective pressures, the set of potential epitopes is precisely the set of all of the HIV 8-10aa peptides. The epitope predictors

for HLA types present in the population can be used to give preference to peptides that are more likely to be epitopes. Note that the whole set of potential epitopes can not be included in the vaccine for practical reasons; rather, the vaccine should include a set which confers protection to a large fraction of the population against a large fraction of incoming viruses.

Are there HIV peptides that can be outright excluded from the set? Presumably, our current understanding of processing and presentation can help us eliminate a peptide that will never be seen by a T-cell. Selective pressures imposed by T-cells direct HIV evolution toward escapes from T-cell detection. The versions of peptides which escaped [60] could be considered for exclusion. There are several candidate mechanisms in the processing and presentation pathway that we could focus on to eliminate a peptide. The cleavage of peptides by proteasomes and immunoproteasomes is a stochastic process, and while both of these complexes have a preference for cleaving in a particular context of amino acids, it is by no means a deterministic process. Frequently the cleavage products overlap [68]. In the case of HIV, known epitopes often come from neighboring offsets in the peptide. An example of overlapping epitopes can be found in the Nef region: the epitopes occur at overlapping offsets 73 (for HLA types A3,A11,B35, A3, B08,B27, B35, C4) ,74 (for HLA types A3,B35,A11) ,75 (for HLA type A101) ,77 (for HLA type B07),79 (for HLA type A02) and 80 (for HLA type A24) [47]. Moreover, even a peptide with an escape mutation for a particular HLA type, may overlap with a viable potential epitope for other HLA types [2]. Hence, a peptide should not be excluded as a potential epitope unless there is strong evidence that it cannot be processed and presented by any of the HLA types in the target human population. An HIV peptide is thus deemed a potential epitope, unless none of the HLA types in the human population recognize it.

Under this definition of potential epitope the vaccine should contain as many of the peptides from the viral population as possible. Note that we cannot guarantee that a potential epitope is indeed an epitope. Hence it is not sufficient to build a vaccine out of a set of peptides such that each virus contains at least one of the peptides, since some of the peptides may not be epitopes. In fact many peptides are expected to not be epitopes. It is impossible to include all

of the HIV peptides in a single vaccine of practical length due to the variability of the HIV and consequently huge number of potential epitopes. Clearly the peptides which can confer protection against larger numbers of viruses should take precedence over the peptides that can provide protection against infrequent HIV strains. Hence, I will assume that the optimal vaccine will contain the most frequent peptides.

4.1 Maximum coverage formulation of vaccine design

To formalize the framework described above we will say that a peptide is covered by a vaccine, if the peptide or an equivalent cross-reactive peptide occur in the vaccine. In order to assess quality of vaccine in this framework, I introduce some functions that help measure how well the vaccine covers a set of peptides. We will say that a string d is covered by a string s if d occurs in string s. A function indicating whether s covers d is

$$\mathrm{occ}(d, s) = \begin{cases} 1, & \text{if string } d \text{ occurs in string } s \\ 0, & \text{otherwise} \end{cases}$$

Given the limited capacity of the vaccine, we prefer to include in the vaccine peptides which elicit an immune response over peptides which are ignored by the immune system. We formalize this preference by associating a weight w^t with every peptide (indexed by t). The weight measures how likely a peptide is to be an epitope. In order to assign this weight, we have to either measure immunogenicity of the peptide or assume a particular model of processing and presentation. Furthermore, this measure can include a bias toward a particular distribution of the HLA types in a population. For example, a peptide that would have a high probability of detection by an HLA type that is common in Europe may actually be mostly undetected if that HLA type is uncommon Africa. In the absence of such a model, a constant weight can be assigned to each epitope.

We will measure the coverage of a set of peptides $D = \{d^1, \ldots d^T\}$ with weights $W =$

$\{w^1, \ldots w^T\}$ by a string s by

$$\mathrm{cvr}(D, W, s) = \text{coverage} = \frac{\sum_{t=1}^{T} w^t \mathrm{occ}(d^t, s)}{\sum_{t=1}^{T} w^t}$$

Strictly speaking, the denominator in the above function is not required for the optimization described later, because weights are fixed; we include it so that cvr has values between 0 and 1.

The problem of finding a vaccine of length N can then be formulated as:

$$vaccine = \underset{s:|s|=N}{\mathrm{argmax}}\, \mathrm{cvr}(D, W, s) = \underset{s:|s|=N}{\mathrm{argmax}} \frac{\sum_{t=1}^{T} w^t \mathrm{occ}(d^t, s)}{\sum_{t=1}^{T} w^t} \tag{4.1}$$

where $|s|$ denotes the length of string s and N is the prespecified length of the vaccine.

When weights associated with peptides are uniform, the coverage equals the fraction of all peptides from the viral population that are present in the vaccine. Ideally, if the immunogencity of peptides is known, the weight of a single peptide should be set to the frequency of HLA types in the human population for which the peptide is an epitope. Obviously, peptides which are not immunogenic for any of the HLA types in the population have weights 0. Given such weights, coverage is a fraction of the expected immunogenicity achieved by the vaccine. The ideal vaccine achieves coverage 1, which means that the maximum expected immunogenicity of the vaccine is achieved.

In the following sections, I will deal strictly with uniform weights; the examination of non-uniform weights is postponed until Section 6.1. One shortcoming of this score is that it does not directly capture cross-reactivity (Section 2.4). For example, either of two epitopes may alone be sufficient to stimulate protection against both of the epitopes. Modifications to the data representation required for dealing with cross-reactivity are addressed in Section 6.2.

In computer science, the problem of finding the shortest string which contains all the strings from a particular set is called the shortest superstring problem. This problem is known to be NP-hard [29]. In order to show that the vaccine design problem is itself NP-hard, I will describe how a polynomial time vaccine optimizer, if one existed, could be used to solve the shortest superstring problem in polynomial time. Given a list of strings for which we wish

to construct one of the shortest superstrings, we place each of the strings into a dataset D and assign to each weight 1. This is a trivial operation and its complexity is linear in the size of the dataset. Let the sum of the lengths of all the strings in the dataset be denoted by Z. For each value x from the range between 1 and Z, we run a separate vaccine optimizer with $N = x$. If a vaccine optimizer achieves coverage 1, then all of the strings are covered and the resulting vaccine is a superstring. When $N = Z$ there exists a trivial solution, concatenation of all the strings, although the vaccine optimizer may return a different solution. Hence we are guaranteed that, at the very least, one of the optimizer runs returns a superstring. Then there exists a minimal N for which the vaccine optimizer returns a vaccine with coverage 1. This vaccine is the shortest superstring. We require at most Z runs of the vaccine optimizer. If each run of vaccine optimization were polynomial in Z then the whole algorithm would be polynomial in Z and we would be able solve the shortest superstring problem in polynomial time. This shows that there exists a polynomial reduction from the shortest superstring problem to the problem of vaccine optimization. Hence the vaccine optimization problem is NP-hard, and since NP-hard problems are not believed to be solvable by polynomial time algorithms, we look for approximate solutions.

4.2 Greedy algorithm

One idea is to first order the peptides according to their weight in descending order, beginning with the first peptide, and starting with an empty vaccine sequence recursively extend this sequence by absorbing additional peptides into the vaccine with as much overlap with the right end of the vaccine as possible. There exist other greedy algorithms which build up separate parts of the sequence, merging onto sequences on both ends, as opposed to the right hand extension used here. However it has been shown that all of these variants, including the version presented here which extends the sequence solely on the right end, are asymptotically optimal [27]. In practice the different variants of the greedy algorithms, including the ones which

progressively merge pairs of strings, provide comparable performance and I describe one of the simpler algorithms.

Let ext(s, d) evaluate to the shortest string s' which contains both string s and string d, such that d occurs at the end of s'. I will refer to the act of composing two strings with maximum possible overlap as extending.

Let ov(s_1, s_2) evaluate to the length of the overlap between the end of the string s_1 and the beginning of string s_2. Directly from the definition, we have $\mathrm{ov}(s_1, s_2) = |s_1| + |s_2| - |\mathrm{ext}(s_1, s_2)|$.

Extending string s by a string d may increase the number of covered strings, and consequently the total weight of covered strings.

Let $\Delta\mathrm{cvr}(s, d, D)$ denote the increase in coverage of a set of strings $D = \{d^1, ..., d^T\}$ by s achieved by extending string s with string d:

$$\Delta\mathrm{cvr}(s, d, D) \quad = \quad \sum_{t=1}^{T} w^t(1 - \mathrm{occ}(d^t, s))\mathrm{cvr}(\{d^t\}, \{w^t\}, \mathrm{ext}(s, d))$$

Input : $D = \{d^1...d^T\}, W = \{w^1...w^T\}$

Output: s

$j = \mathrm{argmax}\, w^i$;

$s = d^j$;

while *true* **do**

  $j = \mathrm{argmax}_i\, \Delta\mathrm{cvr}(s, d^i)$;

  if $\Delta\mathrm{cvr}(s, d^j) = 0$ **then**

    all of the strings have been covered

    return s;

  else

    $s = \mathrm{ext}(s, d^j)$;

  end

end

Algorithm 1: A greedy superstring algorithm.

For input with equal weights, the greedy algorithm (Algorithm 1) approximates a solution to the superstring problem. It has been shown that the greedy algorithm produces a superstring

that is at most 4 times the length of the shortest superstring, and conjectured that the actual upper bound is 2 times the length of the shortest superstring. Alternative approximations that produces superstrings which do not exceed 2.5 times the length of the shortest superstring have been proposed [77].

4.3 An exact method for coverage optimization

In this section, I explore an exact method for coverage optimization. This method allows us to assess approximate methods since it provides the optimal solution for the problem, albeit at extreme computational cost. The basis for this approach is the branch-and-bound technique [81]. I will assume that the optimization problem can be described in terms of binary integer variables and that it is a maximization problem. A branch-and-bound technique recursively builds a tree in which each node is associated with a set of solutions, the root being associated with all possible solutions. The children of the node provide a decomposition of the node's set of solutions. This decomposition is achieved by introducing constraints on the variables, which in the most basic case take the form of variable assignments.

If the tree of solutions is explored naively, for example by a depth-first search algorithm, an optimal solution can be certified only after the whole tree has been explored. However, if we are able to efficiently compute an upper bound on the scores of the set of solutions associated with a node, we can avoid exploring the subtree rooted at this node if any solution with score higher than this upper bound has been found. Since no solution in the subtree rooted at this node could exceed the upper bound and hence cannot exceed the candidate solution's score. Such a candidate solution may originate from a different part of the tree or it may be a heuristic suboptimal solution obtained beforehand. In the case of a binary integer problem, a linear program relaxation can be used to compute an upper bound on the score achievable by the integral solution. Since the set of integral solutions is strictly included in the set of solutions of the relaxation, the score of the optimal solution of the relaxation is at least as great as the score

of the integer program.

The linear program solver may produce a fractional solution that does not satisfy the integrality constraints of the integer program. Alternatively, the integer program may have been relaxed to a linear problem with integer solutions, such as linear programs with unimodular constraint matrices. The solutions of the relaxation will be integral but may violate some of the original constraints of the problem. If none of the constraints are violated and we have an integer solution, then this solution is a candidate for the optimal solution of the original integer program.

A solution which violates the constraints of the original integer problem may be removed from consideration by the introduction of an inequality which holds for the integral solution of the original problem, but not for the solution obtained from the relaxation. Such an inequality is called a cut. A family of techniques which combine the introduction of cuts with branch-and-bound is called branch-and-cut [81].

In the next section, I will give an overview of a variant of the traveling salesman problem, the orienteering problem, which can be seen as an integer program formulation of the vaccine design problem. After that, I will describe a branch-and-cut algorithm for this problem, which is based on the subtour elimination constraints.

4.3.1 The vaccine design problem as an instance of an asymmetric orienteering problem

The orienteering problem [35] is a variant of the traveling salesman problem, which limits the resources available to the salesman for traveling. Given a graph $G = (V, R)$ where V is the set of vertices and R is the set of directed edges, with each vertex v we associate a nonnegative prize p_v and with each edge e we associate a travel time h_e. We will not assume that the travel time is the same in both directions on an edge e. The total travel time is subject to the constraint that it is less or equal to some preset N and we wish to maximize the total prizes collected. Note that it may be possible to collect all of the prizes in less time than the preset N, which is

why we do not use N as an equality constraint on the total travel time.

Now we define a reduction of the vaccine design problem to the asymmetric orienteering problem. The same reduction is used to transform the shortest superstring problem into an asymmetric traveling salesman problem. Each vertex v corresponds to a peptide d^t, the prize associated with a vertex is given by w^t, and the travel time from vertex d^i to d^j is given by $|d^j| - \text{ov}(d^i, d^j)$, which equals the number of amino acids in d^j not covered by the overlap with d^i. Note that this distance is not symmetric. For example, the distance between strings AAA and BAA is 3 since the shortest string which contains AAA at the beginning and BAA at the end is AAABAA. However distance between BAA and AAA is 1, since the shortest string which contains BAA at the beginning and AAA at the end is BAAA.

Given a path that solves the asymmetric orienteering problem, we recover the vaccine V by visiting each vertex in the order given by the path and appending the corresponding strings with overlap. Suppose there existed a vaccine V' with strictly higher coverage than the vaccine obtained from the solution to the orienteering problem. From this vaccine we can construct a path for the orienteering problem, as follows. The orienteering path is initialized to an empty set. Starting with $i = 1$, a check is made to determine if the peptide occurring at offset i of the vaccine matches any of the peptides in the dataset. If there is no such peptide, i is incremented until such a peptide is found. There must be at least one such peptide, since we require coverage of V' to be greater than that of V. Let a denote the index of the peptide found by this step. Again, i is incremented until the peptide at offset i matches some peptide in the dataset that does not occur in vaccine V between offsets 1 and i. Let b denote the index of the peptide found in the dataset. The edge (a, b) is added to the orienteering path and a is set to b. The last two steps are repeated until the end of the vaccine V' is reached. Since by assumption the coverage of the vaccine V' is strictly higher than coverage of the vaccine V, the path we reconstructed from V' must have strictly higher prize total than the original path from which we obtained vaccine V. But this contradicts the assumption of the V's optimality. Hence an optimal solution of the orienteering problem is always the optimal solution of the

vaccine design problem.

A particular solution of the orienteering problem consists of a closed path. The set of vertexes corresponds to indexes of the peptide, and the total amount of prizes collected at each vertex is the sum of the weights of the peptides which were on the path. Note that when the same peptide occurs multiple times in the dataset, this will be reflected in the peptide's weight and consequently in the prize associated with the vertex.

4.3.2 Branch-and-cut method for the asymmetric orienteering problem

Here, I present a branch-and-cut technique similar to the one introduced in [25] for the symmetric variant of the problem. A binary variable x_e is associated with each edge indicating whether the solution path uses this edge. Similarly, y_v is a binary variable indicating whether a particular vertex is on the path. The variable h_e is the length of edge e. We designate the starting and ending node as 1. This is a dummy node that does not correspond to a particular peptide; it has 0 distance to all of the other nodes. Its purpose is to allow us to formulate constraints that uniformly apply to all of the nodes, for example, that each node must have a neighbor before it and a neighbor after it on the path. The linear integer program for our original problem given equation 4.1 is

$$
\begin{array}{lllll}
\max & \sum_{v\in V} p_v y_v & & & \\
\text{s.t.} & \sum_e h_e x_e & \leq N & & (1)\\
& \sum_{v'\in V} x_{(v,v')} & = y_v, & \forall v \in V & (2)\\
& \sum_{v'\in V} x_{(v',v)} & = y_v, & \forall v \in V & (3)\\
& \sum_{v'\in S} \sum_{v''\in V\setminus S} x_{(v',v'')} & \geq y_v & \forall S \subset V, 1 \in S, v \in V\setminus S & (4)\\
& \sum_{v'\in V\setminus S} \sum_{v''\in V} x_{(v',v'')} & \geq y_v & \forall S \subset V, 1 \in S, v \in V\setminus S & (5)\\
& y_1 & = 1 & & \\
& x_e & \in \{0,1\} & \forall e \in R & \\
& y_v & \in \{0,1\} & \forall v \in V &
\end{array}
$$

The inequality (1) ensures that resources spent on travel do not exceed N. In terms of the

vaccine design, the vaccine should not exceed the capacity of the delivery vector (N amino acids) and, in fact, it may be shorter than N if we manage to find a vaccine with optimal coverage that is shorter than N. The inequalities (2) and (3) are the degree constraints which require that a particular vertex is traversed only once. The inequalities (4) and (5) are the subtour elimination inequalities which ensure that every vertex v on the path is reachable from the starting city and that the ending city is reachable from v. In the implementation of the branch and cut technique for the orienteering problem, subtour elimination constraints are added as cuts rather than included in the problem formulation. Note that the set of subtour inequalities is exponential, just as in the traveling salesman problem formulation. The branch and cut techniques avoid inclusion of all of these inequalities, by adding them one at a time when they are violated and only for the subtree for which they are violated. Once that subtree is pruned the cuts are eliminated.

In order to decide which cut need to be added, the most violated constraint is found. For a given node v we can find a minimum capacity cut which separates node 1 from node v. This cut will produce two subsets of nodes: S_v which contains 1 and $V \backslash S_v$ which contains node v. If the maximum flow corresponding to the minimum cut is smaller than y_v then we have found a violated subtour inequality and we can include this inequality to strengthen the linear program. The complexity of finding the most violated subtour elimination inequality is rather high: $O(|V|^4)$ due to the cost of the maximum flow algorithm being $O(|V|^3)$. I resorted to randomly picking a subset of nodes for which to run the max-flow algorithm in order to reduce the running time of the subtour elimination algorithm. In addition, I used the solution of the greedy algorithm as the initial lower bound on the coverage in order to prune the solutions in the branch and bound tree. Due to the significant running time of the algorithm, it was applied only to a small set of sequences used for detailed comparison to the probabilistic methods described below.

4.4 Generative models of peptides

In Section 4.1 I discussed the superstring problem. We sought a string that contained all of the strings from the dataset. Each of the strings could trace its origin to some position in the superstring. Approximately, we could generate the dataset by randomly picking a position in the superstring and copying a short string from that position.

We now formalize this idea in the following fashion. Each position i in the vaccine is associated with a multinomial distribution $E_i(a)$, where a takes values between 1 and 20, corresponding to amino acids. We will call this list of multinomials an epitome, a modeling concept with its origins in computer vision [39]. We assume that we have a probability distribution $p(i)$ ranging from 1 to N, corresponding to the probability of choosing particular position i. Then the probability of generating a peptide d of length L from position i of epitome E is given by:

$$p(d|E,i) = \prod_{k=1}^{L} E_{i+k-1}(d_k)$$

The probability of generating peptide d from some unknown position is:

$$p(d|E) = \sum_{i=1}^{N} p(d|E,i)p(i)$$

and probability of generating a full set $D = \{d^1, ...d^T\}$ of peptides from an epitome E is then given by:

$$p(D|E) = \prod_{t=1}^{T} \left(\sum_{i=1}^{N} p(d^t|E, i^t = i)p(i) \right)$$

This is refered to as the likelihood function for E, in statistics and machine learning

Note that with each peptide d^t we associate a position variable i^t that indicates which position was used to generate the peptide. Even though this position may not be known we include i^t as a hidden variable.

The epitome is assumed to lie on a circle, so that a peptide of length 10aa generated from the very last position of the epitome would be generated from multinomials at offsets $N, 1, 2 \cdot 9$.

This assumption is made to avoid boundary issues. If a peptide with a beginning which overlaps another peptide's end is placed at the beginning of a non-circular epitome, the overlapping peptide would have to be placed before the beginning of the epitome in order to exploit the overlap. This would be impossible in a non-circular epitome. Circularity of the epitome eliminates this problem since there are no boundaries. Note that vaccine inserts are single non-circular sequences, but these sequences can be obtained from an epitome by simply appending the first $L-1$ amino acids to the end of the epitome.

The epitome parameters E can be subdivided into multiple separate lists of multinomials, hence allowing optimization of separate vaccine inserts. In this thesis, I utilize a single list of multinomials for a single vaccine insert.

E can be obtained using maximum likelihood, as described later. By taking the most likely amino acid at each position (i.e., $\operatorname{argmax}_a E_i(a)$), we can obtain a vaccine sequence. An example of a local maximum of the likelihood is an epitome which encodes the shortest superstring, a global maximum of coverage and where $p(i)$ is uniform: $p(i) = \frac{1}{N}$. For a given shortest superstring s, let $E_i(s_i) = 1-\epsilon$ and $\sum_{a \neq s_i} E_i(a) = \epsilon$. The likelihood of a single peptide is then lower bounded by $\frac{1}{N}(1-\epsilon)^L$ where $\frac{1}{N}$ is the contribution from $p(i)$, and $(1-\epsilon)$ is the probability of generating the most likely letter at that position. Since there are T peptides, the likelihood of the whole set of peptides is then lower bounded by $(\frac{1}{N}(1-\epsilon)^L)^T$. Under some circumstances, an optimal solution of the shortest superstring may contain multiple occurrences of the same string. However, as the number of repetitions of the same string is bounded from above by the length of the superstring, the contribution of a particular string to the likelihood is bounded from above by $(1-\epsilon)^L$. Next, I show that the likelihood of the dataset of peptides is maximized if for each peptide there exists at least one position where the peptide is well explained, that is, where epitome assigns high probability to the peptide. Suppose that all of the peptides but one were well modeled by epitome and occurred in multiple locations in the epitome. Furthermore, assume that the peptide which was not well modeled by epitome is mismatched only in one letter. Hence the likelihood assigned to this dataset by epitome would

be upper bounded by $((1-\epsilon)^L)^{T-1}(1-\epsilon)^{L-1}\epsilon$. For a sufficiently small epsilon, the lower bound of the likelihood of an epitome which models well a particular set of peptides exceeds the upper bound of the likelihood of the epitome which models well all peptides but one, but does so in multiple locations. More precisely, $((1-\epsilon)^L)^{T-1}(1-\epsilon)^{L-1}\epsilon \leq (\frac{1}{N}(1-\epsilon)^L)^T$ if $\epsilon \leq \frac{1}{N^T+1}$. Hence a global minimum of the epitome model will always model well all of the peptides, possibly in multiple locations, but never at the expense of not modeling some peptides. Hence, given a sufficient epitome capacity, the shortest superstring is one of the local maxima of the likelihood.

In order to obtain this sequence, I use the EM algorithm [18, 63] to find E which maximizes a lower bound on the likelihood. The EM algorithm is an iterative algorithm which alternates between two steps. The first step, E-step, maximizes the expected likelihood with respect to some form of distribution over hidden variables, such as positions i in the epitome model. The second step, M-step, utilizes the computed distribution over hidden variables, referred to as posterior distribution, to compute updated values of parameters, in our case E. Starting from some estimate of parameters and applying these two steps is guaranteed to produce new set of parameters for which the probability of data is at least as good as it was for the initial estimate of parameters [63]. Hence, iterating these two steps increases the probability of data until convergence. The resulting parameters achieve a local maximum of likelihood. Next, I will introduce a lower bound on the likelihood for the epitome model that can be used to derive an EM algorithm.

$$\log p(D|E) \geq \sum_{t=1}^{T}\sum_{i=1}^{N} q^t(i) \sum_{k=1}^{L} \log\left(E_{i+k-1}(d_k^t)p(i)\right) - \sum_{t=1}^{T}\sum_{i=1}^{N} q^t(i)\log q^t(i) \tag{4.2}$$

where $q^t(i)$ is the current estimate of the probability that peptide d^t occurrs at position i.

Optimization of this lower bound proceeds by iteratively updating the q distributions and the epitome E, as follows.[1]

[1]The updates for distributions in the model, the posteriors q and epitome parameters E, must be normal-

E-step:

$$q^t(i) \quad \propto \quad p(d^t|E,i)p(i) \tag{4.3}$$

M-step:

$$E_i(a) \quad \propto \quad \sum_{l=1}^{L}\sum_{t=1}^{T} q^t(i-l+1)[a=d_l^t] \tag{4.4}$$

$$p(i) \quad \propto \quad \sum_{t} q^t(i) \tag{4.5}$$

I use $[a=b]$ to denote the indicator function which evaluates to 1 if $a=b$ and 0 if $a \neq b$.

4.5 Favoring unique epitope positions and distinct vaccines

The generative model presented in the previous section does not account for two properties that we would want a vaccine to have. First, the epitome may contain multiple copies of the same epitope, which is undesirable because it may lead to wasted space. The objective function we optimize to obtain the ML estimate of the epitome in fact rewards higher entropy in the posterior distribution $q^t(i)$; that is, it favors each epitope appearing at multiple locations in the vaccine or epitome. Second, the ML estimate of the epitome may be ambiguous about the appropriate amino acid to use in some positions. For example, this arises as a result of mapping to the same position i in the epitome of two peptides d^{t_1} and d^{t_2} that differ in their first amino acid. Hence, the updates for the multinomial distribution at the offset i will contain a contribution of $q^{t_1}(i)$ and $q^{t_2}(i)$ of two different amino acids. Since the two peptides are the same in all positions except the first, $q^{t_1}(i)$ and $q^{t_2}(i)$ will be comparable. Assuming that posteriors for other peptides do not favor mapping those peptides to offset i, the contribution of $q^{t_1}(i)$ and $q^{t_2}(i)$ will dominate the update for $E_i(a)$, leading to a mixture of the two amino

ized. Such updates are obtained by taking derivatives of the bound augmented by Lagrange terms, for example $\lambda_i(\sum_a E_i(a) - 1)$. Updates throughout this thesis are given as proportionality equations. These equations are used to compute the unnormalized distributions, which are subsequently normalized to add up to 1.

acids in that position. Since the goal is to obtain a specific amino acid sequence for a vaccine, this ambiguity is problematic.

To address the above desirable properties, we by force the variational posterior $q^t(i)$ to be of the form:

$$q^t(i|\eta^t, v^t) = \begin{cases} \eta^t & \text{if } i = v^t \\ \frac{1-\eta^t}{N-1} & \text{otherwise} \end{cases} \tag{4.6}$$

This form of approximate posterior distribution is called a variational posterior because it cannot be arbitrary, but is instead specified by variational parameters η and v. We can parameterize the epitome E so as to have a peak at a particular amino acid u_i at position i:

$$E_i(a|\theta, u) = \begin{cases} \theta_i & \text{if } a = u_i \\ \frac{1-\theta_i}{|A|-1} & \text{otherwise} \end{cases}$$

where A denotes the alphabet of amino acids, and $|A|$ is the cardinality of set A (i.e., 20). In the rest of this section, I will use θ, u, η and v to denote sets of parameters $\{\theta_1, ..., \theta_L\}$, $\{u_1, ..., u_L\}$,$\{\eta^1, ..., \eta^T\}$ and $\{v^1, ..., v^T\}$ respectively.

By Jensen's inequality we are guaranteed that any distribution $q^t(i)$ in equation 4.2 can be utilized to yield a lower bound on the log-likelihood [63]. Hence we obtain:

$$\begin{aligned} \log P(D) &\geq LB(D; \theta, u, \eta, v) \\ &= \sum_{t=1}^{T}\sum_{i=1}^{N} q^t(i|\eta, v) \left(\log p(i) \prod_l E_{i+l-1}(d_l^t|\theta, u) \right) - \sum_{t=1}^{T}\sum_{i=1}^{N} q^t(i) \log q^t(i) \\ &= \sum_{t=1}^{T}\sum_{i=1}^{N} q^t(i|\eta, v) \log p(i) + \sum_{t=1}^{T}\sum_{i=1}^{N}\sum_{l=1}^{L} q^t(i|\eta, v) E_{i+l-1}(d_l^t|\theta, u) - \\ &\quad \sum_{t=1}^{T}\sum_{i=1}^{N} q^t(i) \log q^t(i) \end{aligned}$$

The tightness of the bound depends directly on parameters θ, u, η, v. Importantly, for an epitome derived from the shortest superstring with all θ_is and η^ts being 1, the bound is tight.

Hence the form of the posterior and the new parametrization do not eliminate the local minimum we seek.

The lower bound on the log-likelihood can be maximized with respect to its parameters in order to obtain "a peaked epitome":

$$\log p(D) \geq \max_{\theta,u,\eta,v} LB(D;\theta,u,\eta,v)$$

As in the previous section, we subdivide updates into two steps, which are described below. The first step is the variational E-step which updates parameters η and v of the variational posteriors $\{q^t\}$ by solving optimization problem $\max_{\eta,v} LB(D;\theta,u,\eta,v)$ for fixed θ and u obtained in the previous iteration. Note that we can independently update each pair of η^t and v^t, given that θ and u are fixed . The second step updates the θ and u parameters of the epitome by solving the optimization problem $\max_{\eta,v} LB(D;\theta,u,\eta,v)$ for the fixed η and v obtained in the last iteration.

4.5.1 Derivation of the variational E-step

In order to obtain the updates, we first expand the objective function $LB(D;\theta,u,\eta,v)$ by plugging in the form of the variational posterior and the epitome parametrization:

$$LB = \overbrace{\sum_t \sum_i \eta^{t[i=v^t]} \left(\frac{1-\eta^t}{N-1}\right)^{[i\neq v^t]} \left(\log p(i) + \sum_l \log E_{i+l-1}(d_l^t)\right)}^{\text{expected log-likelihood}} \underbrace{- \sum_t \sum_i \eta^{t[i=v^t]} \left(\frac{1-\eta^t}{N-1}\right)^{[i\neq v^t]} \log\left(\eta^{t[i=v^t]} \left(\frac{1-\eta^t}{N-1}\right)^{[i\neq v^t]}\right)}_{\text{entropy term}}$$

Note that the second part of the expression, the entropy term, can be further simplified to

$$\sum_t \eta^t \log \eta^t + \sum_t \left(1-\eta^t\right) \log \frac{1-\eta^t}{N-1}$$

and does not depend on the value of v^t.

To maximize LB wrt v^t and η^t, we could iterate over all possible settings for v^t and compute the corresponding value of η^t. Comparing LB for each of the v^t, η^t pairs, we could obtain their optimal values. However, this is unnecessary since we can obtain the best v^t without iterating over all possible values, as follows. Each summand in the expected log-likelihood term can be subdivided into two subterms denoted a and b below.

$$\sum_i \underbrace{\eta^{t^{[i=v^t]}} \left(\frac{1-\eta^t}{N-1}\right)^{[v^t \neq i]}}_{a} \underbrace{\left(\log p(i) + \sum_l \log E_{i+l-1}(d_l^t|\theta, u)\right)}_{b}$$

A trivial application of the rearrangement inequality[2] applied to the term guarantees optimality of the update

$$v^t = \underset{i}{\operatorname{argmax}} \left(\log p(i) + \sum_l \log E_{i+l-1}(d_l^t)\right) = \underset{i}{\operatorname{argmax}} \left(p(i) \prod_l E_{i+l-1}(d_l^t)\right) \tag{4.7}$$

under the condition that $\eta^t > \frac{1}{N}$, i.e. the "peaked" letter has to have the highest probability assigned to it . We will show later that updates of η^t maintain this condition. The above update is in line intuitively with our goal of obtaining a peaked posterior.

For the optimal v^t we can obtain the update for η^t by taking the derivative of $LB(D; \theta, u, \eta, v)$ with respect to η^t and setting the derivative to 0. After simplification, the following update for η^t is obtained:

$$\eta^t = \frac{p(v^t) \prod_l E_{v^t+l-1}(d_l^t)}{(N-1)\left(\prod_{i \neq v^t} p(i) \prod_l E_{i+l-1}(d_l^t)\right)^{\frac{1}{N-1}} + p(v^t) \prod_l E_{v^t+l-1}(d_l^t)} \tag{4.8}$$

Note that the choice of v^t guarantees that

$$\left(\prod_{i \neq v^t} p(i) \prod_l E_{i+l-1}(d_l^t)\right)^{\frac{1}{N-1}} \leq p(v^t) \prod_l E_{v^t+l-1}(d_l^t)$$

[2]The rearrangement inequality $a_1 b_1 + a_2 b_2 + \cdots + a_n b_n \geq a_1 b_{\sigma(1)} + a_2 b_{\sigma(2)} + \cdots + a_n b_{\sigma(n)}$ holds for any permutation σ of the set $\{1, \ldots \text{n}\}$ if $a_1 \geq a_2 \geq \cdots > a_n$ and $b_1 \geq b_2 \geq \cdots \geq b_n$

for all $i \neq v^t$. Hence, the numerator of the update is bounded from above by

$$N\left(p(v^t)\prod_l E_{v^t+l-1}(d_l^t)\right)$$

and the update maintains $\eta^t \geq \frac{1}{N}$, which we required for optimality of the update of v^t.

4.5.2 Derivation of the variational M step

The entropy term does not depend on the parameters of the epitome θ and u. The part of the expected likelihood term that depends on position k in the epitome can be written in the following form:

$$\sum_{c\in A}\sum_t\sum_{i,l:i+l-1=k}\underbrace{q^t(i)[d_l^t=c]}_{a}\underbrace{\log\left(p(i)\theta_k^{[u_k=c]}\left(\frac{1-\theta_k}{|A|-1}\right)^{[d_l^t\neq c]}\right)}_{b} \tag{4.9}$$

Again by application of the rearrangement inequality we obtain the optimal update for u_k:

$$u_k = \operatorname*{argmax}_c \sum_t \sum_{i,l:i+l-1=k} q^t(i)[d_l^t=c] \tag{4.10}$$

under the condition that $\theta_k \geq \frac{1}{|A|}$, which will be guaranteed by our initialization and subsequent updates of θ_k. Note that the values of i in the above sums range over a small set of values $\{k-L,...,k\}$, and that for each of those is there exists a unique l. Hence each sum is composed of only L summands.

The updates of θ_k are given by:

$$\theta_k = \frac{\sum_t \sum_{i,l:i+l=k} q^t(i)[d_l^t=u_k]}{\sum_t \sum_{i,l:i+l=k} q^t(i)[d_l^t=u_k] + (|A|-1)\sum_t \sum_{i,l:i+l=k} q^t(i)[d_l^t\neq u_k]} \tag{4.11}$$

The algorithms are initialized with near uniform distributions for E. In the case of "peaked" epitome this means that each u_i is chosen randomly, and θ_is are initialized near $\frac{1}{A}$; that is, a small positive random value uniformly chosen between 0 and 0.001 is added to each θ_i. In the case of fully parameterized epitome, for each value a, $E_i(a)$ is initialized with $\frac{1}{A}$, a small

positive random value uniformly chosen between 0 and 0.001 is added to each $E_i(a)$, and the distribution E_i is normalized. This addition of noise in the initialization of epitome parameters is important, since it breaks symmetry between positions in the epitome. If the symmetry is not broken, i.e. the epitome is initialized with uniform probabilities $\frac{1}{A}$ for all letters, the posterior q^t also becomes uniform, leading to a poor local minimum where each position in the epitome is assigned the marginal distribution of occurrence of amino acids in the peptides.

4.6 Garbage models

The vaccine problem formulated as maximum coverage (Section 4.1) requires the length of vaccine s to be equal to some preset value N. Limitations on the length may make it impossible to cover all of the peptides in the dataset. The generative model presented in Section 4.4, on the other hand, attempts to model all of the peptides. In the extreme case, where the length of the vaccine is equal to the length of a single epitope, the ML estimate of the vaccine in the plain epitome model will be the average of all of the epitopes, whereas the desired solution is a vaccine consisting of the most frequent epitope. While the constraints on the posterior alleviate this kind of problem, these constraints are not sufficient, since they only ensure that each peptide is mapped well at some position in the epitome. In the case of a short epitome, there may not be sufficient capacity to map all of the peptides. Simply put, the epitome must attempt to model all of the data well, even if it is not feasible to do so. In this section I tackle the problem by changing the model rather than constraining the posteriors.

Given a limited capacity of the epitome, it could only have generated a portion of the peptides in the dataset with high probability. Even though the epitome can generate any peptide, the vaccine, which is a sequence rather than a distribution over sequences, can generate only a small set of peptides. Since we use the mode of the epitome as the vaccine sequence, the peptides generated from the vaccine sequence are exactly peptides which are assigned high probability by the epitome. We prefer that the epitome explain a subset of peptides nearly

perfectly while the rest of the peptides are explained by a different distribution. Hence a better generative model is that of a mixture of an epitome and a "garbage" model. We will use π to denote the probability of selecting the epitome to explain a peptide and $1-\pi$ is the probability of selecting the garbage model. Then,

$$p(d^t|E,\pi) = \pi p(d^t|E) + (1-\pi)g(d^t)$$

One possible choice of $g(d)$ is $g(d) = 20^{-L}$. This garbage model is used in experiments in this thesis.

An optimal coverage sequence, given limited capacity preventing inclusion of all peptides, is a shortest superstring s' for a subset of peptides. This sequence corresponds to a local maximum of the likelihood for fixed π and garbage model $g(d^t)$ since the epitome constructed from the string s' ($E_i(s'_i) = 1-\epsilon$ and $\sum_{a\neq s'_i} E_i(a) = \epsilon$) is a local maximum of the epitome term $\prod_t \pi p(d^t|E)$ when ϵ tends to zero and the garbage model assigns uniform probability to all patches, so the garbage likelihood term only depends on the number of discarded patches.

It may be possible to derive the exact EM updates rather than resorting to deriving updates for a lower bound. However, since we want to explore the impact of the restrictions discussed in the previous section, we seek to derive more modular updates, i.e. updates that can be used if the posterior $q^t(i)$ is not exact. If we use an exact posterior, the bound is tight and the updates for $q^t(h)$ become updates of the exact EM.

For learning, in order to obtain a lower bound on the log-likelihood, we introduce a hidden variable h^t associated with peptide d^t which chooses the epitome $p(d^t|E)$ or the garbage model $g(d^t)$. We will use $h^t = 1$ to indicate that the peptide d^t is generated by the epitome, and $h^t = 2$ to indicate that the peptide is generated by the garbage model. The lower bound on log-likelihood is then given by:

$$\begin{aligned}\log p(d^t|E,\pi) \;\geq\; & \sum_t \sum_h q^t(h) \sum_i q^t(i) \sum_l \log \left(\left(\pi p(d^t|E,i)\right)^{[h=1]} \left((1-\pi)g(d^t)\right)^{[h=2]} \right) \\ & - \sum_t \sum_h q^t(h) \log q^t(h) - \sum_t \sum_i q^t(i) \log q^t(i)\end{aligned}$$

The update for the posterior probability of a data point being generated by the epitome ($h = 1$) is derived by taking a derivative of the lower bound with respect to $q^t(h = 1)$. Prior to taking the derivative we simplify the bound by observing that the garbage model's contribution to the likelihood does not depend on hidden variable i. Hence the bound is given by:

$$\begin{aligned}\log p(d^t|E,\pi) \;\geq\; & \sum_t q^t(h=1)\left[\sum_i q^t(i)\log\left(\pi p(d^t|E,i)\right)\right] + \\ & \sum_t q^t(h=2)\left[\log\left((1-\pi)g(d^t)\right)\right] \\ & -\sum_t\sum_h q^t(h)\log q^t(h) - \sum_t\sum_i q^t(i)\log q^t(i)\end{aligned}$$

Taking derivatives with respect to $q^t(h = 1)$ and $q^t(h = 2)$ we obtain the following updates:

$$q^t(h=1) \propto e^{\sum_i q^t(i)\log\left(\pi p(i)p(d^t|E,i)\right)} \quad q^t(h=2) \propto (1-\pi)g(d^t) \tag{4.12}$$

The result of this modification is that the epitome parameters' update is also changed by substituting each $q^t(i)$ with $q^t(h = 1)q^t(i)$. For example, the new update for the unconstrained epitome is:

$$E_i(a) = \sum_{l=1}^{L}\sum_{t=1}^{T} q^t(h=1)q^t(i-l+1)[a = d_l^t] \tag{4.13}$$

Updates for the constrained epitome are similarly modified.

A number of algorithms can be derived from the presented updates. Algorithm 2 provides a template for building algorithms based on the variety of EM updates. It is possible to change updates between different iterations. In the experiments in this thesis, the updates were the same throughout each run of the algorithm.

Input : $D = \{d^1 ... d^T\}$, $W = \{w^1 ... w^T\}$,MaxIter,N

Output: s

for *i=1:L* **do**
Initialize E_i with near uniform distribution

end

for *Iter=1:MaxIter* **do**
E-step:

for *t=1:T* **do**
Compute $q^t(i)$ using equation 4.3 or using equations 4.6, 4.7 and 4.8;

Optionally compute $q^t(h)$ using equations 4.12 and multiply each $q^t(i)$ with $q^t(h = 1)$

end

M-step:

for *i=1:N* **do**
Update E_i using equation 4.4 or using equations 4.9, 4.10 and 4.11

end

end

for *i=1:N* **do**
$s_i = \text{argmax}_a \, E_i(a)$

end

Algorithm 2: Template for the epitome algorithms.

Chapter 5

Results

In this chapter, I first formulate eight distinct EM algorithms based on the updates presented in the previous chapter. These eight algorithms are compared to exact and greedy algorithms on a small dataset, where the size of the dataset is dictated by the limitations of the exact algorithm. The best method is then compared in a coverage test to the synthetic vaccine design methods, cocktails of consensus sequences, cocktails of centers of tree, and random strain cocktails. The same methods are then evaluated on a population protection test under different assumptions about the link between coverage and protection. This link is given as the number of viral peptides which must be present in the infecting virus and the vaccine in order for the immunized patient to clear the infection. Finally, the coverage optimized vaccines are validated using wet-lab experiments, and I estimate the level of protection based on these experiments.

5.1 Comparison of vaccine design methods using coverage

A number of methods can be developed by including or excluding various constraints and modifications outlined above, which include:

1. Parameterizing the epitome so that there is a peak on a particular amino acid and uniform distribution on other amino acids (Peaked Amino Acid)

2. Parameterizing the posterior so that there is a peak on a particular position of a peptide in the epitome and a uniform on other positions (Peaked Position)

3. Including a garbage model (Garbage)

Each option in the above list can be either included or excluded to produce a method. This yields a total of eight possible combinations. I will refer to methods that do not restrict the epitome parametrization as the Full Amino Acid methods. Similarly, methods that do not restrict the posterior over positions the Full Position methods. Using this terminology a Full Amino Acid Full Position method is the exact EM algorithm outlined in Section 4.4.

In addition to the EM methods outlined above, I will also evaluate the performance of the following cocktail methods (for details see Section 3.3):

1. Clustering - Consensus method, clustering with a single center [66], and cocktail of consensuses, clustering with multiple centers (see Section 3.3.1)

2. COT - center of tree [64], and generalization with multiple centers of tree (see Section 3.3.1)

3. Strains from data - randomly chosen strains from the Perth dataset

As I pointed out in Section 3.3.1, the clustering and COT methods produce vaccines which are equal to $K \times AL$, where K is the number of vaccine components, and AL is the length of an aligned sequence. Since alignment of HIV sequences is typically produced with respect to HXB2 sequence, AL usually equals the length of the corresponding region in HXB2. Note that the strains method does not use aligned sequences. This method chooses multiple strains from the viral sequences in the dataset, hence the length of the vaccine produced by the strains method may not be exactly an integral multiple of AL. Note that for Gag region $AL = 501$ and for Pol region $AL = 1004$, coverage for cocktail methods will be evaluated at integral multiples of these values for vaccines based on these regions.

In order to compare the coverage of all of the 8 EM methods on the problem of finding a vaccine, I applied all of the methods to a subset of 20 Gag sequences randomly extracted from the HIV sequence dataset [60]. The choice of the size of the dataset was dictated by the run time of the exact method. The number of peptides (T) in this dataset was 9682. The length of each peptide (L) was 10 amino acids. In all cases, the length of the vaccine was fixed to 4 Gag regions of the reference sequence HXB2 [46], 2004 amino acids in total. The results are summarized in Table 5.1.

It might seem that the best coverage should be obtained by including all of the modeling and posterior restrictions. However, in Table 5.1 we see that, in general, Peaked Position degrades performance. This is due to the averaging of posterior over the positions imposed by the parametrization, which effectively lowers the posterior probability of peptide origination from the several top positions. This leads to an inability to switch the choice of the peaked position after the first couple of iterations, making the algorithm heavily dependent on a good initialization. The best coverage is obtained with the combination of Full Amino Acid, Full Position and Garbage. The second best coverage is achieved with a combination of Peaked Amino Acid and Full Position.

Given that the Peaked Position uses an approximation to the true position posterior, we expect the likelihood bound to be worse than the one produced by the algorithm which uses the true posterior, for example the exact EM from 4.4. This degradation in likelihood bound translates into degradation of coverage. Similarly, the Peaked Amino Acid uses a parametrization which yields a lower bound on the likelihood given by the Full Amino Acid parametrization, and we would expect worse performance overall by the Peaked Amino Acid parametrization. However, in cases without the garbage model the Peaked Amino Acid parametrization compensates for the attempts to explain all of the data and produces better coverage than the Full Amino Acid. The restricted parametrization of the Peaked Amino Acid forces a particular amino-acid to dominate at each position in the epitome. This leads to closer approximation of the coverage score by the likelihood function. Hence we obtain a better coverage score with

Table 5.1: Comparison of coverage results for a combination of model parameterizations and posterior restrictions. The first line in each box is the mean (standard deviation shown in parentheses) of coverage (in percent) over 10 runs of each algorithm. The second line contains the maximum coverage over the 10 runs. The greedy algorithm (see Section 4.2) achieved coverage of 63.26%. The exact algorithm (see Section 4.3.2) achieved coverage of 74.43%

	Peaked Position and Peaked Amino Acid	Peaked Position and Full Amino Acid	Full Position and Peaked Amino Acid	Full Position and Full Amino Acid
without garbage	39.24 (6.36) max:50.65	23.88 (3.82) max:27.61	58.91 (2.64) max:62.61	54.82 (2.79) max:58.91
with garbage	51.85 (3.77) max:56.30	48.73 (3.06) max:52.83	52.61 (5.24) max:55.65	63.48 (2.96) max:68.04

Peaked Amino Acid.

The exact method offers best coverage, 74.43%, but it comes at extraordinary computational cost of approximately 54 hours of computational time compared to approximately 0.21 hours for 10 random initializations of a single epitome method on the same dataset. Note that the Full Position Full Amino Acid method with Garbage achieves and impressive maximum of 68.04% near the coverage of the exact method.

A further comparison of the methods is achieved by directly comparing the performance of each of the algorithms when the same initialization is used for all algorithms. In order to achieve the same initialization for all of the methods, the E-step in the first iterations for each of the algorithms was computed by the full epitome E-step and the resulting posterior $q^t(i)$ provided to M-step corresponding to different algorithms. Each of the following iterations were performed using E-step and M-step for the corresponding algorithm. For the initial E-step, the full epitome was initialized by the same random assignment where each of the multinomial values was perturbed away from 0.05 by the addition of a uniform noise between 0 and 0.001. The multinomials were then renormalized. The results of this comparison are given in Table 5.2.

Table 5.2: Mean and standard deviation of difference of coverage starting from same random initialization for different methods over 20 different random restarts. Methods used indicated by the following: FP - Full Position; PP - Peaked Position; FA - Full Amino Acid; PA - Peaked Amino Acid; G - garbage model. Each entry is formed by subtracting the coverage result of the method corresponding to the row from the results of the method corresponding to the column. The bolded entries show the change in coverage when the garbage model is included.

	PP PA	PP FA	FP PA	FP FA	G PP PA	G PP FA	G FP PA	G FP FA
PP PA		15.36 (5.22)	-19.67 (5.82)	-15.58 (5.67)	-12.61 (7.47)	-9.49 (6.77)	-13.37 (7.59)	-24.24 (6.02)
PP FA	-15.36 (5.22)		-35.04 (2.87)	-30.94 (3.62)	-27.97 (3.17)	-24.86 (3.86)	-28.73 (7.81)	-39.60 (3.24)
FP PA	19.67 (5.82)	35.04 (2.87)		4.09 (4.11)	7.07 (3.25)	10.18 (2.44)	6.30 (5.68)	-4.57 (2.80)
FP FA	15.58 (5.67)	30.94 (3.62)	-4.09 (4.11)		2.97 (3.70)	6.09 (4.37)	2.21 (6.66)	-8.66 (2.39)
G PP PA	**12.61 (7.47)**	27.97 (3.17)	-7.07 (3.25)	-2.97 (3.70)		3.12 (3.03)	-0.76 (7.35)	-11.63 (2.33)
G PP FA	9.49 (6.77)	**24.86 (3.86)**	-10.18 (2.44)	-6.09 (4.37)	-3.12 (3.03)		-3.88 (4.97)	-14.75 (3.46)
G FP PA	13.37 (7.59)	28.73 (7.81)	**-6.30 (5.68)**	-2.21 (6.66)	0.76 (7.35)	3.88 (4.97)		-10.87 (6.30)
G FP FA	24.24 (6.02)	39.60(3.24)	4.57 (2.80)	**8.66 (2.39)**	11.63 (2.33)	14.75 (3.46)	10.87 (6.30)	

Employing the garbage model improves results in all cases but one: Peaked Position and Full Amino Acid. The garbage model slows down the convergence in all cases (see Figure 5.1). However, the garbage model coupled with Full Position and Full Amino Acid does give the best maximum coverage and the best mean coverage. The coverage achieved by this model improves over the greedy by 68.04-63.26=4.78%.

The coverage optimization methods introduced so far can produce a vaccine of any desired length. Clearly, shorter vaccines will have lower coverage.

The methods compared have diverse running times. By far the fastest algorithm is the clustering method, which is my generalization of the previously published consensus method [66]. The Center of Tree method requires fitting of the phylogenetic tree; this is the most time-consuming part of this method. The epitome EM algorithms perform a number of iterations, each of which takes polynomial time; however, there are no guarantees that the algorithm converges in a polynomial number of such iterations. In the experiments, the EM algorithm was interrupted when improvement in likelihood fell below a preset threshold. The branch-and-cut method is solving an NP-hard problem exactly, in possibly exponential time. In order

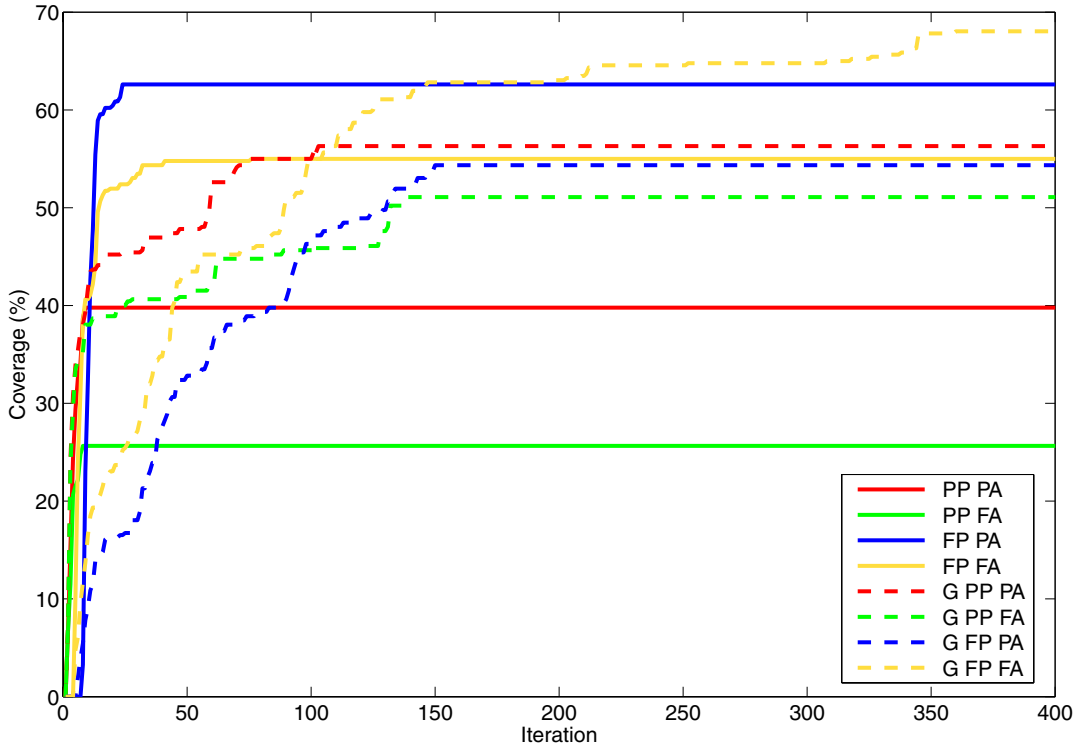


Figure 5.1: Best coverage achieved for each iteration of the EM algorithm for different methods on the dataset of 20 Gag sequences. Methods used indicated by the following: FP - Full Position; PP - Peaked Position; FA - Full Amino Acid; PA - Peaked Amino Acid; G - garbage model. A particular choice of posterior form and epitome parametrization corresponds to a color, while presence of the garbage model is denoted by a dashed line and its absence by a solid line. Inclusion of the garbage model slows down the convergence of the EM algorithm, but ultimately can lead to better coverage.

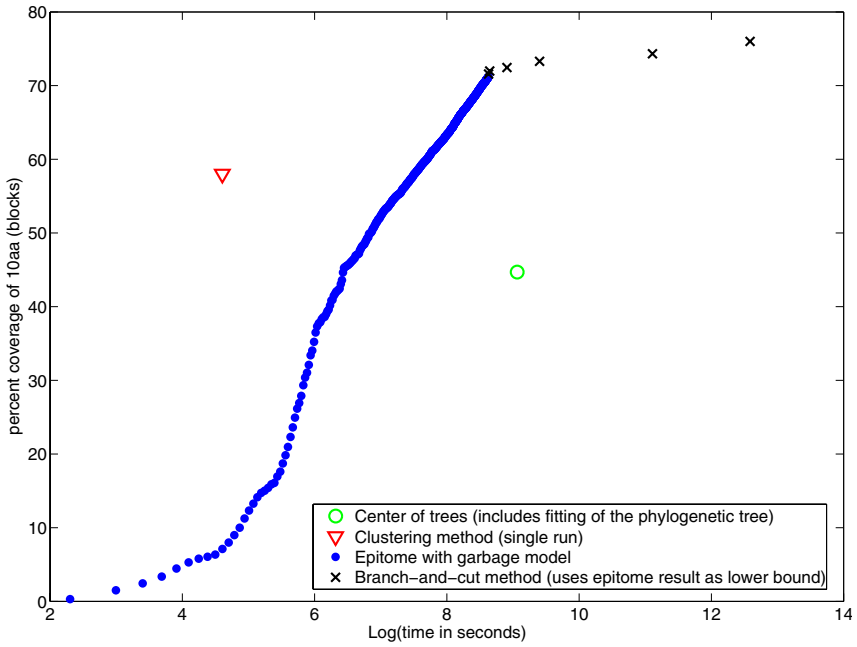


Figure 5.2: Running times of different algorithms on logarithmic scale. Clustering and Center of Tree methods output a single solution at the end of the run; hence they are represented with a single datapoint. For the epitome EM algorithm, coverage at the end of each iteration is plotted. The exact, branch-and-cut, method utilizes coverage obtained by the EM algorithm as a lower bound on the optimal coverage, hence its timing starts once the epitome algorithm is done. For each integral solution obtained by the branch-and-cut method that does not violate any of the constraints of the problem, coverage and the time at which it was obtained are plotted.

to facilitate more efficient pruning, the coverage result of the epitome run is used as a lower bound on the coverage when starting the branch and cut algorithm. Running times on the set of 20 Gag sequences extracted from the Perth dataset are provided in Figure 5.2.

Next, I present the coverage results on the full Perth sequence dataset [60], which contains 245 HIV sequences (Figure 5.3). I compare coverage results of epitome, Full Posterior Full Amino Acid with Garbage with 20 restarts, against three other methods: clustering, center(s) of tree (COT), randomly selected strains. These methods are described in Section 3.3.1. Note that for the last three methods, we present only the results at integral multiples of the length of

a region for which vaccine is designed, Gag or Pol. The reason for this is discussed in 3.3.1. In addition, the strains method need not have exactly the length of a multiple of the region, as the collected sequences in the Perth dataset vary in length and are not directly aligned to HXB2.

We see that epitome achieves coverage that is 5-10% higher than all previously published methods. Note that the for the shorter capacity epitome the coverage differences are smaller. In particular, for vaccines of the length of a single virus, roughly all of the methods have equal performance. This is suggestive of difficulty of coverage optimization with short vaccines. The problem seems to become easier for our techniques when more capacity is provided.

A particular vaccine which matches a virus in a single potential epitope may elicit a T-cell response which offers protection against that virus. A single epitope is sufficient for protection against a non-variable virus, but multiple epitopes are required to clear a rapidly changing population of viruses. In Figure 5.4 I show the number of infections which can be cleared by an immune system primed by Gag vaccines of varying length. A vaccine is deemed unable to clear the infection unless the original infecting virus contained a minimum number of peptides also occurring in the vaccine, and this minimum number was varied to see its effect. In the six plots, the minimums I used were a) 5 b) 25 c) 50 d) 100 e) 175 and f) 200 peptides. The infecting viruses were taken from the Perth dataset [60]. This experiment illustrates that the epitome has not only the most frequent peptides, but also sufficient breadth to offer protection against a number of distinct viruses. Note that the presence of a single peptide that is present in the vaccine and a virus is not sufficient for protection, since the peptide may not be an epitope (i.e., may not be presented and learnt by the immune system). However with a requirement of only 5 peptides (Figure 5.4a), all vaccines can protect against any virus. In the high end of the spectrum starting with requirement of 100 peptides (Figure 5.4d), we start to see separation between the different vaccines. For comparison the average distance between known epitopes [47] for the same HLA supertype in Env region is 132.35 amino acids in Env region and 29.57 in the Nef region. Hence the plots 5.4c) and 5.4d) indicate potential performance difference between vaccines.

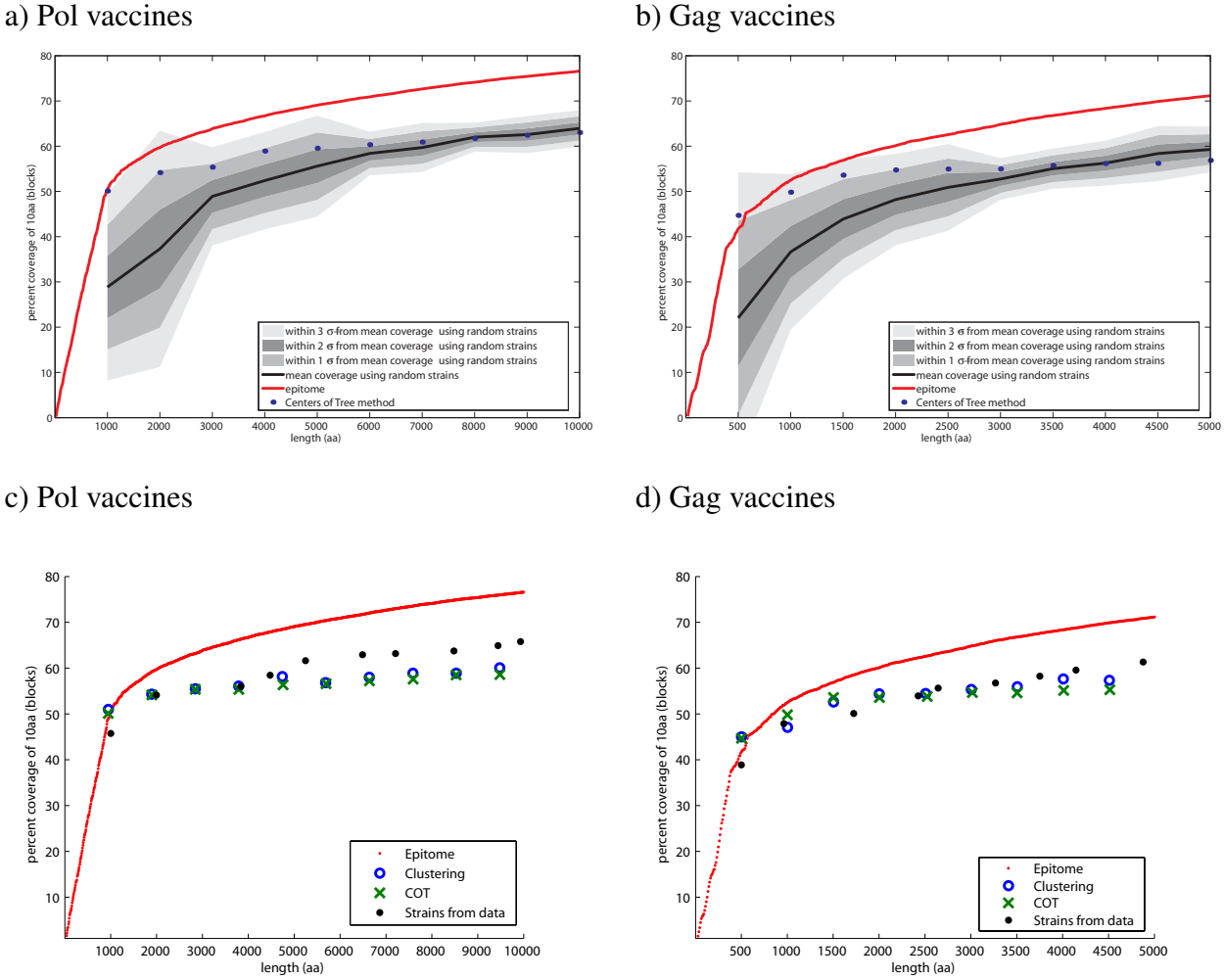
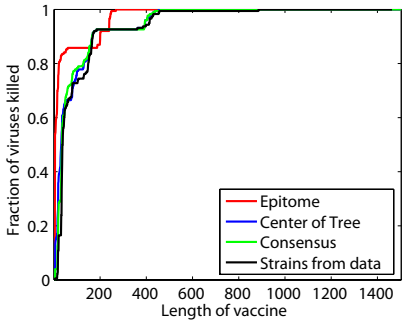
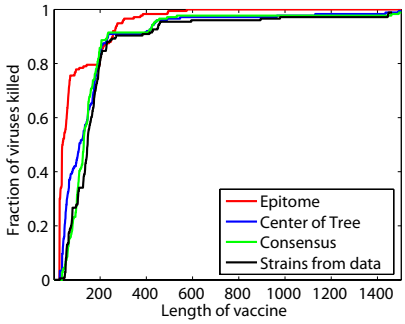


Figure 5.3: Coverage of uniformly weighted 10aa long peptides from the Perth dataset. Comparison between an epitome (using garbage model, full Position, full epitome parametrization) and the Center of Tree method (COT), Hamming distance clustering and randomly selected strains. The last three methods produce vaccines which are integral multiples of the length of the corresponding region (501aa for Gag, 1004aa for Pol), hence for these methods coverage is only evaluated at those points, ie. for Gag 501, 1002, 3006 etc. and Pol 1004, 2008, 3012 etc. Note that the random strains method sometimes produced vaccines that did not exactly match up with the integral multiple of the length for the corresponding region. This is due to variability in the length of sequences in the Perth dataset. The same issue does not affect Clustering and COT methods since the same Perth sequences were aligned to HXB2. For plots a) and b) each of the randomly selected strains experiments has been repeated 20 times and resulting means and standard deviations are given in the plot. In the plots c) and d) only the best randomly selected strains results are reported along with the results for the other three

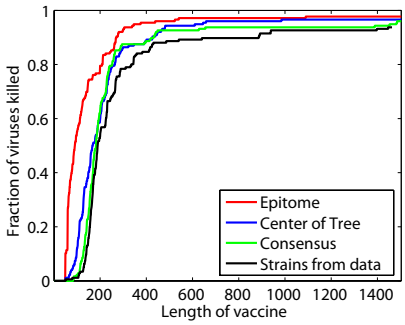
a) 5 peptides needed to clear infection



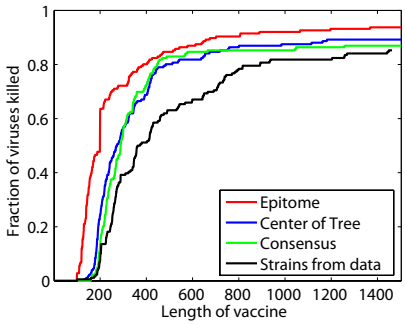
b) 25 peptides



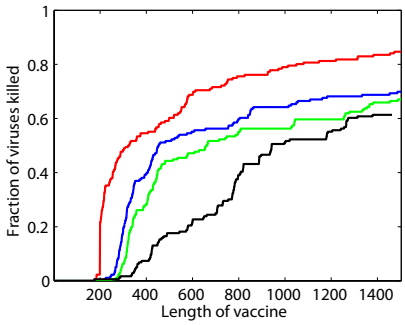
c) 50 peptides



d) 100 peptides



e) 175 peptides



f) 200 peptides

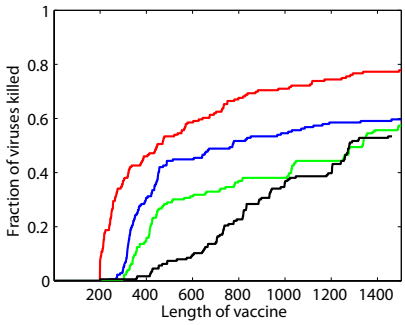


Figure 5.4: The fraction of virus infections cleared by an immune system primed with Gag vaccines of varying lengths. In these experiments, an immune system is deemed able to clear an infection if it was trained by a vaccine which matches the infecting virus in more than a) 5 b) 25 c) 50 d) 100 e) 175 and f) 200 peptides . Note that this does not mean that each of these peptides is an epitope. Rather, the larger the number of peptides in both vaccine and virus, the more likely that they share an epitope as well. If the vaccine has fewer than the prerequisite matches to the virus, it is assumed that an insufficient number of epitopes is available in the vaccine, and the virus is either not detected or it can accumulate escape mutations to evade the

5.2 Wet-lab validation of vaccine based on optimizing coverage

In order to assess the viability of coverage-based vaccines, as well as compare performance of the algorithms, three vaccines were derived for the Nef region of the HIV virus, and tested in Brander lab at Harvard . The first vaccine was the consensus sequence for the nef region. The second vaccine, 3-strains, contained three wild type HIV strains which had highest combined coverage of 10aa peptides in the population of HIV viruses. The last vaccine was optimized purely for coverage of the peptides 10aa and designed using the epitome algorithm for coverage optimization outlined in previous sections. This vaccine was of the same length as the second vaccine - 3 times the length of the Nef region. All of the overlapping 10aa peptides from each of the three vaccines were synthesized. ELISPOT experiments are generally conducted on longer peptides which may contain optimal epitopes as a substring. For example, the study in [21] reported that 18 amino acid long peptides with 10 amino acid overlap are sufficient for discovering epitopes, hence any of the peptides contained in the 18mer may contain optimal epitopes. Here it is also assumed that 8, 9 or 10 amino acid long epitopes inside the 10aa long peptides can be recognized. For each of the peptides, 29 different ELISPOT assays were conducted. The Peripheral Blood Mononuclear Cells (PBMCs) used in the assays were extracted from HIV positive patients, HLA types of all of the patients are given in Table 5.3. An ELISPOT assay indicating a reaction between a T-cell and a peptide indicates that the patient has encountered the particular peptide and was able to mount a response. Hence the peptide was processed and presented by an MHC type I molecules corresponding to one of the patients' HLA types. Table 5.4 lists the number of peptides that different vaccine candidates recognized by each of the patients.

Given that a particular patient is able to react to a set of peptides, we can ask if the patient immunized by each of the three different vaccines could clear an infection by a population of HIV viruses. In this scenario the 29 patients approximate a human population. I used 245 Nef

patient	patient's HLA types						patient	patient's HLA types					
1	A-02	A-30	B-53	B-58	C-04	C-05	16	A-02	A-33	B-35	B-57	C-04	C-07
2	A-01	A-74	B-1503	B-5301	C-02	C-04	17	A-02	A-03	B-35	B-4901	C-07	C-1601
3	A-02	A-11	B-38	B-44	C-12	C-1601	18	A-02	A-32	B-1518	B-44	C-05	C-07
4	A-23	A-36	B-53	B-58	C-04	C-06	19	A-2902	A-6801	B-40	B-52	C-020202	C-150201
5	A-02	A-36	B-27	B-53	C-04	C-15	20	A-23	A-66	B-40	B-44	C-04	C-08
6	A-2301	A-6601	B-4201	B-5301	C-06	C-17	21	A-02	A-11	B-35	B-40	C-03	C-04
7	A-24	A-32	B-15	B-51	C-08	C-1602	22	A-02	A-03	B-08	B-44	C-05	C-07
8	A-3303	A-74	B-1503	B-5801	C-02	C-07	23	A-11	A-11	B-35	B-51	C-04	C-15
9	A-03	A-74	B-35	B-57	C-04	C-07	24	A-02	A-33	B-1503	B-440301	C-02	C-16
10	A-02	A-32	B-14	B-39	C-07	C-08	25	A-0202	A-3001	B-1516	B-4201	C-14	C-17
11	A-02	A-32	B-14	B-39	C-07	C-08	26	A-02	A-03	B-15	B-35	C-030301	C-04
12	A-02	A-24	B-07	B-44	C-05	C-0702	27	A-02	A-30	B-15	B-42	C-14	C-17
13	A-02	A-31	B-35	B-51	C-04	C-1601	28	A-68	A-68	B-07	B-08	C-07	C-07
14	A-23	A-32	B-18	B-8101	C-07	C-08	29	A-0301	A-1101	B-0702	B-3901	C-0702	C-1203
15	A-03	A-31	B-39	B-44	C-05	C-12							

Table 5.3: HLA types of 29 HIV positive patients used in the experimental validation of vaccines

.

patient	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
consensus	22	2	4	4	11	5	**1**	13	8	8	12	8	17	18	19
3-strains	39	**7**	7	4	**21**	10	**1**	23	**14**	12	16	**27**	30	38	27
epitome	**47**	4	**10**	**6**	**21**	**11**	**1**	**29**	**14**	**20**	**26**	**27**	**31**	**39**	**31**

patient	16	17	18	19	20	21	22	23	24	25	26	27	28	29	
consensus	22	6	4	4	6	12	18	12	5	9	3	5	38	7	
3-strains	44	**18**	**8**	4	**18**	**35**	**32**	**32**	**11**	14	**15**	**9**	82	11	
epitome	**47**	15	6	**8**	15	33	31	31	8	**15**	10	8	**83**	**15**	

Table 5.4: Number of epitopes in different vaccine candidates recognized by HIV positive patients. Maximal entries in each column are bolded.

sequences from the Perth dataset to approximate the HIV viral population. Hence there are 245 times 29 different events, corresponding to each patient's exposure to one of the viruses from the Perth dataset. I assumed that all of the epitopes in the vaccine recognizable by a particular patient can be learned by the immune system. For now, I ignore issues raised by immunodominance, but these are addressed in a later section. Furthermore, I assume that a single epitope is sufficient to clear infection by the initial infecting virus. Note that in the previous section I assumed that having some minimum number of peptides occurring in both the vaccine and the virus guarantees that at least one epitope would be found among these shared peptides. Here, in this section, the existence of an epitope is guaranteed by the experimental results. The secondary immune response is quite rapid; it can be mounted within hours of infection, whereas the reproductive cycle of HIV takes a couple of days. Hence it is plausible that a primed immune system may clear the initial infection before the HIV virus manages to discover an escape. Under these assumptions, an immune system primed by a consensus vaccine can counter 55.50% of infections, a 3-strain primed immune system 80.79% of infections and the epitome vaccine primed immune system 89.35% of infections. Hence the two coverage optimized vaccine outperform the consensus vaccine, which is currently considered state-of-the-art in HIV vaccine trials (see Section 3.3). In addition, the epitome vaccine outperforms the 3-strain vaccine in this scenario. The above analysis does not take into account cross-reactivity, but the effects of cross-reactivity would only increase the number of successful clearings of infection. Table 5.5 shows the fraction of each of the infection events, a combination of a virus and a patient, which can be cleared under different hypotheses of how many epitopes need to occur in both the vaccine and the virus in order for the immune system to clear the infection. Note that the 3-strain method provides higher protection than an epitome-based vaccine under the assumption that the minimum number of epitopes required for protection is between 5 and 8. One reason for this is that the 3-strain vaccine, by virtue of each of its components being an HIV strain, has higher potential for breadth, i.e. it has peptides from each of the HXB2 positions, whereas epitome simply focuses on high coverage peptides. In cases where breadth as

well as coverage are required, as is the case when increasing the minimum number of required epitopes, the 3-strain vaccine performs better.

Under the assumption that there exists a strict limit on the number of epitopes that can be acquired, R, by a particular patient, and that a single epitope is sufficient to detect an infected cell, we can compute the efficacy of the 3 Nef vaccines. For any given patient, a vaccine may contain a number of reactive epitopes in excess of the number of acquired epitopes R. In such cases, R epitopes are randomly selected from the set of reactive epitopes for a patient, and the patient is assumed capable of learning only those epitopes. Given the sets of epitopes that each patient can learn, the level of protection for each patient vaccinated by a particular vaccine is given by their ability to detect infecting viruses from the Perth dataset. This experiment was repeated 50 times for each R from 4 to 12, and for each of the three vaccines. Table 5.6 contains the results of this evaluation. Each entry is given as a mean with standard deviation in the parenthesis. The coverage optimized vaccines offer better protection than consensus sequence on this test as well. The epitome and 3-strain vaccines provide comparable protection at $R = 4$, but for all $R \geq 5$ the epitome vaccine offers significantly higher protection.

min number of epitopes	consensus	3-strain	epitome
1	55.50%	80.79%	**89.35%**
2	21.31%	68.37%	**70.50%**
3	8.46%	51.92%	**54.96%**
4	4.41%	40.21%	**41.20%**
5	1.52%	**30.82%**	29.20%
6	0.68%	**24.21%**	22.25%
7	0.27%	**18.66%**	16.33%
8	0.17%	**12.33%**	11.54%
9	0.11%	8.59%	**9.30%**
10	0.00%	5.46%	**7.04%**
11	0.00%	4.62%	**5.77%**

Table 5.5: Fraction of infection events, a (virus,patient) pair, which can be cleared if the patient is immunized by one of the three vaccines. Each row corresponds to a hypothesis of the minimum number of epitopes recognizable by the patient which need to occur in the infecting virus in order for the primed immune system to clear infection by the virus. Note that unlike in Figure 5.4 we are guaranteed that know which peptides are epitopes and hence we do not have to use large number of peptides shared between virus and vaccine as a proxy for presence of an epitope.

R	consensus	3-strain	epitome
4	32.53(2.25)%	51.18(4.15)%	50.86(4.80)%
5	37.96(2.02)%	54.16(2.43)%	61.09(2.88)%
6	41.73(2.35)%	59.92(1.88)%	65.83(2.20)%
7	44.19(1.95)%	66.20(1.74)%	69.69(1.55)%
8	45.58(2.14)%	68.55(3.04)%	75.64(1.70)%
9	47.84(2.26)%	70.88(1.78)%	77.48(1.49)%
10	49.56(1.90)%	72.21(1.28)%	80.07(1.49)%
11	50.28(1.19)%	73.82(1.34)%	81.71(1.73)%
12	51.78(0.77)%	75.50(0.94)%	82.89(1.48)%

Table 5.6: Fraction of infection events, a (virus,patient) pair, which can be cleared if the patient is immunized by one of the three vaccines. It is assumed that the minimum number of epitopes recognizable by the patient which need to occur in the infecting virus in order for the primed immune system to clear infection by the virus is 1. Each row corresponds to a hypothesis about the maximum number of epitopes R which can be acquired from a vaccine. For each entry 50 experiments were performed in which each of the patients who can react to more than R epitopes in the vaccine was assigned a random subset of R epitopes learned from the vaccine. Patients who can recognize fewer than R epitopes were assigned all of the epitopes they can recognize. The protection levels were then computed using the assigned set of epitopes. The mean and standard deviation, in parenthesis, over these 50 experiments are reported in each entry.

Chapter 6

Possible extensions to this work

In this chapter, I will illustrate possible extensions to the algorithms which optimize pure coverage. The first sections describe how predictions from models of immunological processes, such as epitope predictors, can be included in the vaccine optimization. The second section shows how a simple model of cross-reactivity can be included in the vaccine optimizer and its impact on the resulting vaccine. The third section demonstrates how a dataset of peptides can be extended with predicted peptides which may evolve from existing HIV sequences. The fourth section deals with design of vaccines which utilize multiple coding frames. The last section examines possible challenges to vaccines based on coverage optimization, including the various extensions of the first four sections. This section describes problems of immunodominance, protein folding of synthetic vaccines and autoimmune response to vaccines and how these two problems may be addressed in the vaccine design.

6.1 Incorporating models of immunological processes

A number of models of processing, presentation and immunogenicity have been proposed (see Section 3.4). This section aims to illustrate how such models as well as models of effects such as cross reactivity can be incorporated into the vaccine optimization framework outlined above. The first part of this section deals with elimination of sampling bias inherent in treating each of

the peptides as immunogenic, given a model of immunogenicity. Second part illustrates impact of such an immunogenicity model on coverage optimization. The last part of this section deals with incorporation of a simple cross-reactivity model in the vaccine optimization framework.

6.1.1 Sampling bias and models of immunological processes

The generative model introduced above can be used to model a set of HIV derived peptides. By construction, each of the peptides in an HIV strain is present in dataset D. The processing and presentation in a cell generates a different set of peptides. Under the assumption that the cell can generate all the peptides in original set D, albeit with non-uniform probability, we can correct for the bias in the uniform sampling scheme we used to derive set D.

We approached the problem of obtaining a vaccine as a maximum likelihood estimation. Supposing that the true distribution of the peptides generated in a cell is given by f and the empirical distribution of peptides we obtained from the HIV strains is $\hat{f}$, we wish to maximize the likelihood of a dataset derived from f rather than $\hat{f}$.

$$\begin{aligned}\sum_d f(d)\log p(d|E) &= \sum_d \frac{f(d)}{\hat{f}(d)}\hat{f}(d)\log p(d|E) \\ &\approx \sum_t \frac{f(d^t)}{\hat{f}(d^t)}\hat{f}(d^t)\log p(d^t|E)\end{aligned}$$

Hence, if we obtain samples from distribution $\hat{f}$ we need to reweight each of the samples with $\frac{f(d^t)}{\hat{f}(d^t)}$. For each peptide d^t, we set a corresponding $w^t = \frac{f(d^t)}{\hat{f}(d^t)}$.

Since $\hat{f}$ is uniform we only need to weight contribution of each of the data points in the EM algorithm above, resulting in update

$$E_i(a) \propto \sum_{l=1}^{L}\sum_{t=1}^{T} q^t(i-l+1)[a = d_l^t]\frac{f(d^t)}{\hat{f}(d^t)}$$

This correction allows us to introduce priors on presentation and processing which can be obtained independently of the vaccine construction.

6.1.2 Utilizing a model of immunogenicity

In this section, I explore the impact of an immunological prior on the vaccine design. In order to take advantage of the immunological prior, we reweight, as described in section 6.1.1, each of the peptides with $h(d|hla)p(hla)$, where $h(d|hla)$ is the probability that peptide d is an epitope for HLA type hla, and $p(hla)$ is frequency of the HLA type hla in the target population. The distribution h is obtained by running the algorithm from [41] on each of the peptides. We utilize the HLA distribution derived from the [60] dataset. In Figure 6.1 I compare the performance of the epitome EM algorithm optimizing the weighted coverage to different vaccines which do not use the weights. There are two epitome vaccines: one optimized for pure coverage and another optimized for weighted coverage. Both are scored on weighted coverage. The rest of the vaccines are optimized according to their respective scores, but scored both on pure coverage and weighted coverage.

Note that weighted coverage is generally slightly higher, due to low expected immunogenicity. However even the epitome optimized for pure coverage still manages to provide a reasonable weighted score, in spite of being optimized without regard to immunogenicity. This implies that for a given peptide with low immunogenicity there frequently exist overlapping peptides with high immunogenicity. In turn, inclusion of the high immunogenicity peptides often includes the low immunogenicity at no extra cost in terms of space, due to overlap.

6.2 Simple cross-reactivity model

Systematic characterization of the cross-reactivity of epitopes would have tremendous impact on vaccine design. Broad cross-reactivity of epitopes would allow us to reduce the number of peptides that need to be included in the vaccine. In turn, vaccines which avoid redundant peptides could cover more of the true variability of the virus. The coverage score, described

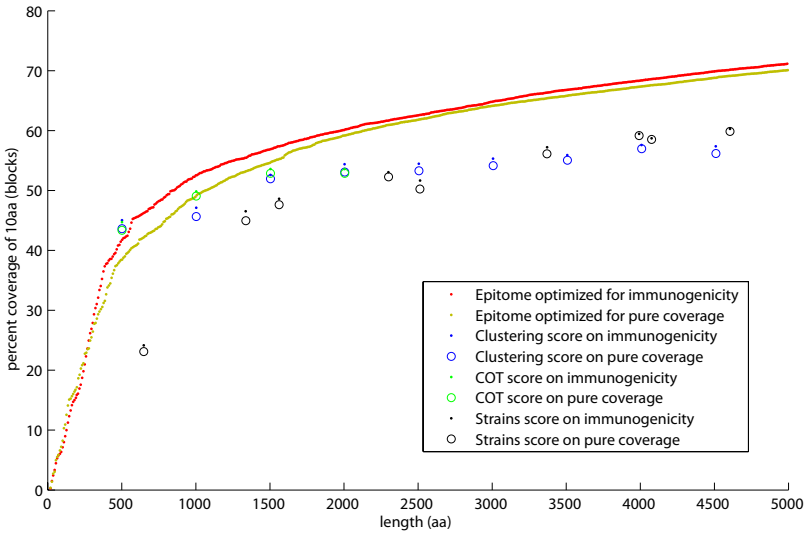


Figure 6.1: Performance of different methods on weighted vs. unweighted coverage score. The weights represent the expected immunogenicity of a particular epitope and were derived from an epitope predictor and distribution of HLA types in the target population.

in Section 4.1 is a conservative score since it assumes no cross-reactivity. I will show how the coverage score can be modified to account for cross-reactivity. Expression $A \sim B$ will denote that two peptides A and B are cross reactive.

In order to accommodate the cross-reactivity we redefine function occ introduced in Section 4.1 as:

$$\mathrm{occ}(d, s) = \begin{cases} 1, & \text{if any of the strings } d' \in CR(d) \text{ occurs in string } s \\ 0, & \text{otherwise} \end{cases}$$

where $CR(d) = \{d' | d \sim d'\}$.

The likelihood function is redefined as $p(d|E) = \sum_{d'} p(d'|E, d' \sim d)p(d' \sim d)$. Note that $p(d' \sim d)$ is the probability of cross-reactivity between the peptides d and d'. The simple model of cross-reactivity used in this section will be a deterministic function, but optimization with an uncertain model of cross-reactivity is the same. One critical requirement is that evaluation $p(d|E)$, as defined above, is tractable.In other words, the sum over all peptides d' which cross-

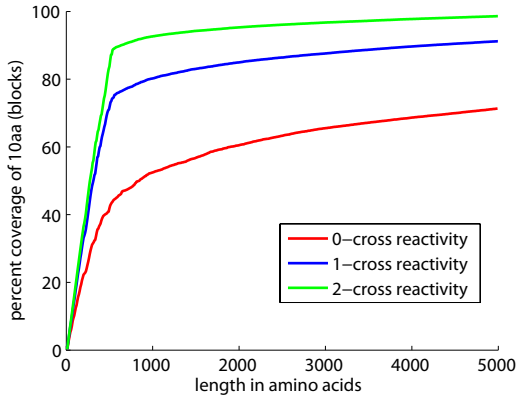


Figure 6.2: Coverage of uniformly weighted 10aa long peptides from the Perth dataset under different cross reactivity models, where two peptides k-cross react if they differ in up to k positions. Each curve corresponds to a vaccine optimized for coverage under a particular cross reactivity model.

react with peptide d has to be computable in time polynomial in the length of the peptide. Note that the space of all 10 aa long peptides is quite large, 20^10, and naive evaluation of the above sum would be expensive.

I will use a simple model of cross reactivity which considers two epitopes cross-reactive if they differ in up to k offsets. I will say that the peptides k-cross react. Two peptides which k-cross-react will k'-cross-react for all $k' \geq k$. For a given k Expression, $A \sim B$ will denote that peptides A and B k-cross-react. Hence, 0-cross reactivity actually assumes no cross-reactivity, and L-cross reactivity means that all of the 20^L possible peptides are equivalent.

Here I compare on a naive cross-reactivity model the impact of breadth of cross-reactivity on vaccine design. The model of cross-reactivity is especially simple: an epitope A is said to k-cross react with an epitope B if they differ in at most k positions. A model where there is no cross reactivity is an 0-cross reactivity model.

Using, as in the previous section, the Gag sequences from the Perth dataset, I compare the coverage of sequences produced by the algorithm under different cross-reactivity models.

Most strikingly, the 2-cross reactivity model implies that a short vaccine with large coverage could be achieved in little over 2 times the length of the Gag region. This indicates that the 2-cross reactivity model is unlikely to be biologically sound. We expect that the coverage achievable under a biologically accurate cross reactivity model would fall somewhere between the 0-cross reactivity and 1-cross reactivity model. The reason for this is that the 1-cross reactivity model implicitly assumes that all mutations have the same small impact on presentation. Hence, if one variant is immunogenic, all of the 1-cross reactive peptides are as well. However, not all of the positions with an epitope are equivalent in their importance in the presentation pathway. A mutation in the anchor residue may abolish presentation of a particular epitope. However, the coverage difference between the two curves suggest that including a cross-reactivity model may result in significant reduction in the length of the vaccine.

6.3 HIV evolution in presence of vaccine

HIV quickly evolves to avoid immune pressure of a naive immune system. Can an immune system primed with a coverage optimized vaccine mount a strong response and clear the infection? An immune system primed with the coverage optimized vaccine would eliminate some of the avenues of escape. Given a model of evolution we can evaluate how likely any given HIV virus is to find an escape from immune system trained by the coverage optimized vaccine. With current estimates of an HIV generation time, from infection of to production of a virus which and subsequent infection of another cell, is approximately 2.6 days [70]. During this replicative cycle, the step of noisy reverse transcription of viral RNA introduces changes in the viral genome. The reverse transcribed DNA will be used to generate all of the viruses budding off of the infected cell. Since the human transcription machinery makes fewer mistakes, the viruses built by a single infected cell all carry the same RNA. Only after one of these new viruses infects a cell and starts performing reverse transcription will there be a chance for another mutation.

Following the first replicative cycle approximately 10^9 new virions are created. The point mutations occur at a rate 10^{-5} per base per replication cycle [12]. As the result viral population can escape in a single epitope after the first replication cycle since it will have 10^4 viruses with appropriate mutation. Each subsequent generation will contain at least one virus attempting each of the possible escape in additional 1.8 epitopes. Given rapid response of the memory T-cells, the response is mounted within hours of infection [48], the infection may be cleared by response from memory cells targeting single epitope prior to the end of the reproductive cycle of the virus.

The above rough analysis makes an assumption of that each of the mutations is uniform. However not all escapes are equally likely, due to their impact on viral viability. In the rest of the section I utilize a PAM250 matrix derived from the Perth dataset in order to construct vaccines which attempt to block likely avenues of escape. Hence we enlarge the dataset of peptides with single point mutated peptides. Each of the added peptides is weighted by the probability of it occurring after the first replicative cycle.

The graph shows the coverage of peptide set which includes both HIV peptides and hypothesized escapes under the PAM250 matrix. The set of peptides is significantly enlarged compared to the original set of peptides. It includes 20 times more peptides since none of the peptides are excluded, rather they may have small weights under the evolutionary model. Figure 6.3 illustrates effects of including the single point mutated peptides in the training set. Sequence optimized only for coverage of original HIV peptides has significantly lower coverage than the sequence optimized for the coverage of the extended dataset. Note that the extended dataset overestimates variability of the HIV viruses, since some of the mutations may not produce viable viruses.

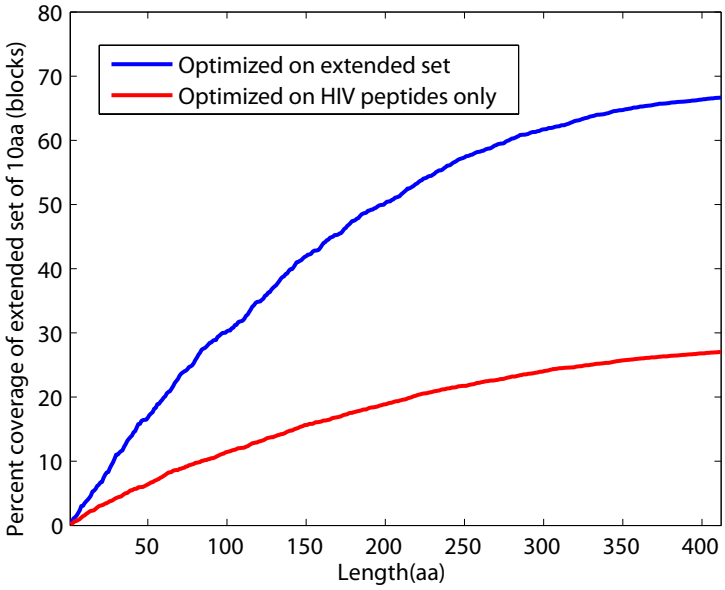


Figure 6.3: Coverage of set of HIV peptides from Nef region extended with single point mutated peptides. Each peptide is weighted by probability of mutation under PAM250 matrix and frequency of the original HIV peptide. The two sequences are compared, one optimized for coverage of the extended set and another optimized for coverage of HIV peptides only.

6.4 Using multiple frames for a vaccine

Under immune pressure, viruses evolve rapidly, producing many different variants adapted to different persons. In [60] it was shown that mutations at many sites in the HIV-1 virus correlate with the patients' HLA type. In Figure 6.4 I show two epitopes in the reference sequence, HXB2. In individual patients, there can be significant variability in the viral peptides corresponding to the same HXB2 position. For example, in data from [60] the peptide TLWQRPLVT in Pol occurs in 72% of the patients. Another variant, TLWQRPLVS occurs in the same position in 8% of the patients, and the patients may have numerous other variants, e.g., TLWQRPLVN, TLWQRPLVI, TLWQRPLVA, TLWQRPIVT, as well. The other epitope emphasized in Figure 6.4 is the Gag epitope YPLTSLRSL, which is present in 13% of the patient population I analyzed, and is only the third most frequent peptide at that position. The more frequent variants are YPLASLRSL which is found in 32% of the patients, and YPLASLKSL, present in 23% of the analyzed population. There are a total of 5 variants of this peptide that have a frequency of more than 5%, and the total number of different variants found among 247 patients was over 20. When diversity in different regions of the genome is combined, the combinatorial explosion leads to the large variability of HIV.

In contrast to this immense diversity of HIV-1 viruses is the tendency of HIV to keep its genome short. This is achieved by utilizing the same nucleotide content to encode different proteins in three different frames (see Figure 6.5). An even more extreme example is the case of the Hepatitis B virus: the sequence encoding gene S is wholly contained in the Pol gene, only in a different frame (Figure 6.6). Given that such utilization of multiple frame occurs in nature, we may want to exploit this in designing compact vaccines.

6.4.1 Multiple frame vaccine optimization

In Figure 6.5 I illustrated the structure of the HIV genome. HIV genes are found in multiple overlapping frames. A particular DNA sequence allows six different amino acid coding frames,

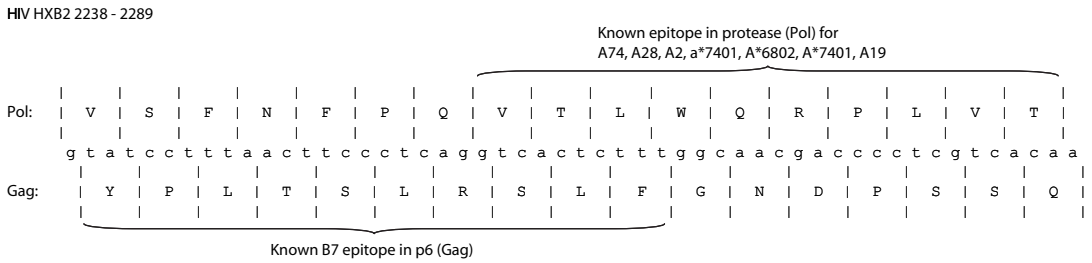


Figure 6.4: An example of two known HIV epitopes in two different coding frames.

comprised of 3 frames in one direction along the DNA ("sense") and 3 frames in the other direction ("anti-sense"). We aim to take advantage of the efficient packing of Gag and Pol regions that HIV already uses.

The modifications to the generative model and vaccine are straightforward (see Figure 6.7) . The vaccine, or rather the epitome is now a list of 3N multinomials over 4 possible letters corresponding to nucleotides. The length of the epitome is extended to accommodate the same length in amino acids. A particular peptide is generated by choosing a position in the epitome and a direction. The starting position gives us the corresponding frame. The direction, either positive or negative, indicates sense and anti-sense strands. A sequence of nucleotides is generated from epitome and translated into amino-acids, resulting in a peptide.

More formally, the probability of generating a peptide d is given by:

$$p(d|E) = \sum_i \sum_{b\in\{-1,+1\}} p(i,b) \prod_{l=1}^{L} \sum_{(n_1,n_2,n_3)\in\mathcal{N}^b(d_l)} E_{i+b3l}(n_1) E_{i+b(3l+1)}(n_2) E_{i+b(3l+2)}(n_3)$$

Where $\mathcal{N}^{+1}(a)$ is a set of possible nucleotide triplets (n_1, n_2, n_3) that encode for amino acid a (Recall that multiple nucleotide triplets can encode the same amino acid). The variable b indicates whether the triplet should be flipped to account for the sense and anti-sense strands. For example, amino acid Y can be encoded as TAT or TAC in sense; alternatively, in the anti-sense direction, it can encoded as ATA or GTA . Note that the sequence GTA is obtained by reversing the complement of TAC. The set of complemented encodings in the anti-sense

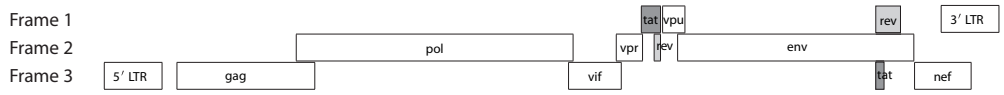


Figure 6.5: Map of the HIV genome across three coding frames. Gag is encoded in Frame 3 and its end reaches past the starting nucleotide of Pol which is coded in Frame 2. Since the two genes are in different frames, the shared nucleotide sequence is translated into two different amino acid sequences.

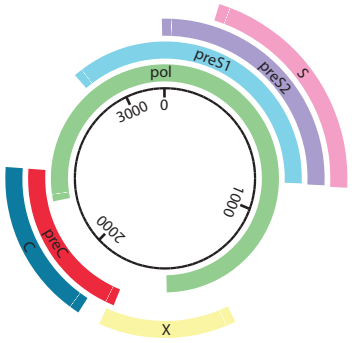


Figure 6.6: The genome of Hepatitis B virus. The Pol gene contains the S gene. The two genes are expressed in different frames.

direction is given by $\mathcal{N}^{-1}(a)$. As in the previous sections, i denotes a position in the epitome.

Given this formulation, the following E-step update is obtained simply by computing the probability of $p(d|E)$ and renormalizing the posterior:

$$q^t(i,b) = \alpha \sum_i \sum_{b\in\{-1,+1\}} p(i,b) \prod_{l=1}^{L} \sum_{(n_1,n_2,n_3)\in\mathcal{N}^b(d_l^t)} E_{i+b3l}(n_1) E_{i+b(3l+1)}(n_2) E_{i+b(3l+2)}(n_3)$$

where α is computed so that $\sum_i \sum_b q^t(i,b) = 1$.

The M-step update for the epitome parameters is given by

$$E_i(n) = \beta \sum_t \sum_j \sum_{b\in\{-1,1\}} \sum_{r=1}^{3} \sum_{l=1}^{L} \sum_{(n_1,n_2,n_3)\in\mathcal{N}^b(d_l^t)} q^t(j,b)[j+((l-1)*3+r-1)*b = i][n_r = n]$$

where β is computed so that $\sum_n E_i(n) = 1$.

The indicator function $[j+((l-1)*3+r-1)*b=i]$ is meant to ensure that the nucleotide at position i of the epitome is used for encoding the r^{th} nucleotide of the codon for the l^{th} amino acid of peptide d^t.

6.4.2 Multi frame vaccine in-silico examination

I used a dataset of HIV sequences from [60] collected from 247 different individuals with chronic HIV infection in Western Australia. From these sequences, I obtained decoded Pol and Gag proteins. These proteins were chosen because they are coded in two different frames with overlap of about 207 nucleotides. I focused on 66 nucleotides corresponding to positions 2227 to 2292 in the prototypical HIV-1 sequence, HXB2 [46]. I determined the amino acid sequences that are coded in the regions of overlap for both Pol and Gag. From these subsequences, I produced a set of all 9-mers with overlap, which was then pruned to remove duplicate 9-mers. The resulting set contained 220 distinct 9-mers.

I ran three different algorithms on this dataset to produce three candidate vaccines. These candidates were scored and compared on coverage of 9-mers: the fraction of all 9-mers that can be found either encoded in the candidate vaccine, in the case of nucleotide sequences, or occurring exactly, in the case of amino acid sequences. A choice of a particular encoding for amino acid should not impact immune system response, since the immune system detects foreign peptides. The same algorithms can be applied to optimize the coverage of epitopes, rather than all 9-mers, once all, or at least most, HIV epitopes have been found for a particular host/virus population. Note that the lengths of all vaccine candidates were truncated to a preset length in nucleotide units. In the experiments below, I used lengths of 180, 300 and 600 nucleotides.

The first algorithm (Random Strains) randomly chooses a number of original HIV sequences, extracts the nucleotide region under study and concatenates these regions together to produce a vaccine candidate.

The second algorithm (Single Frame) operates directly on amino-acids of the 9-mers; it is

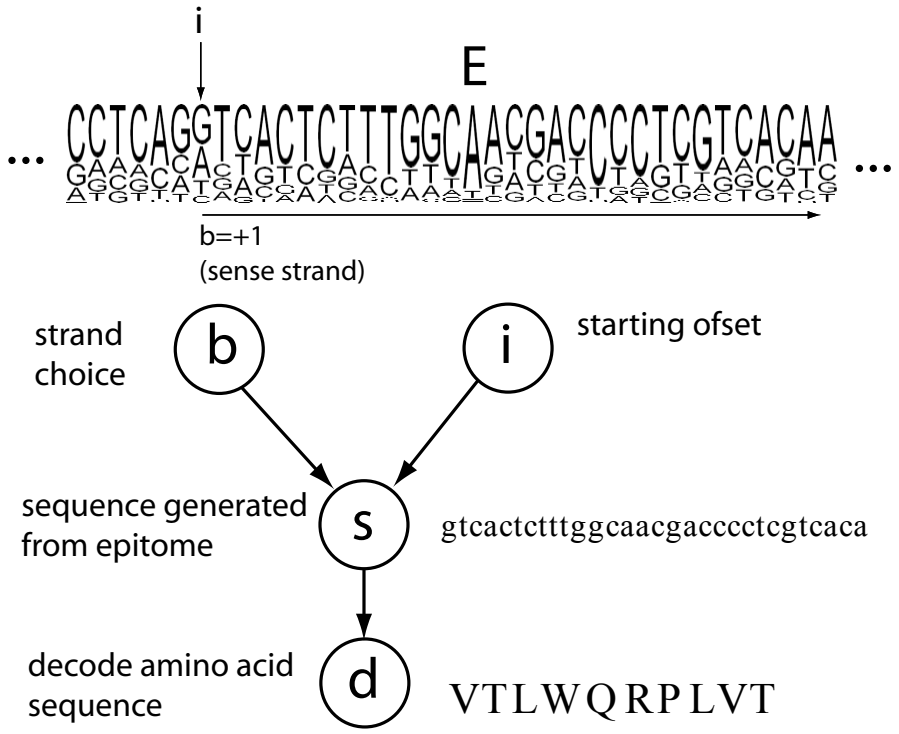


Figure 6.7: The multiframe epitome as a generative model. An offset i and direction b are chosen. These two variables fully determine one of the 6 possible frames. If direction b is positive, then the nucleotide sequence s is generated from the epitome from positions i through $i + 26$. If direction b is negative, a nucleotide sequence is generated from positions i through $i - 26$ and complemented to obtain sequence s. The amino acid sequence d is obtained by translating triplets of nucleotides into amino acids.

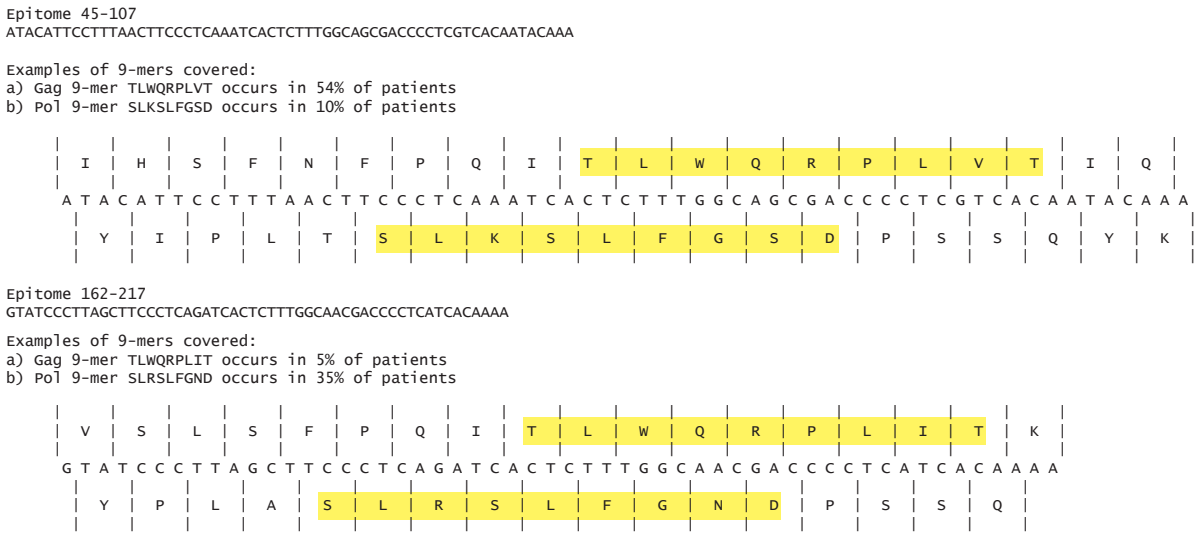


Figure 6.8: The multiframe epitome captures the diversity of HIV by learning various versions of each region, with targeted overlapping peptides corresponding to the different amino acid coding frames. While this particular epitome was optimized for packing (having high likelihood) of 9-mers in the data, it automatically packs variants of the two epitopes illustrated in Figure 6.4 together with their context necessary for presentation. For example, the entire 16-long pattern YPLASLRSLFGNDPSSQ shown above is present in 24% of patients. In general, the vaccine optimized for coverage of all overlapping 9-mers typically contains long stretches from the original strains, and thus should undergo similar peptide processing in the cell as the viral proteins it was trained on. However, as opposed to strain cocktails, epitomes are constructed to maximize the number of different epitopes they contain, and avoid unnecessary repetitions of less variable region. This is achieved because epitome can model all occurrences of the less variable region with a single copy of it, whereas in strain cocktail each strain contains a copy of the region.

the greedy algorithm presented in Section 4.2. The vaccine candidate is constructed by greedily appending (with overlap) 9-mers until a given length is achieved. Note that this algorithm produces an amino acid sequence, unlike the first algorithm which works in the nucleotide domain.

While the first algorithm utilizes multiple reading frames to the extent that the original HIV sequences do, the second algorithm makes no attempt to do so. Indeed, it can be considered to perform optimization in a single frame, since the overlap occurs on the amino-acid level.

The third algorithm (Multiframe) is based on maximizing the likelihood of the given set of 9-mers under the model described in the previous section. The learning is performed by the EM algorithm. In order to initialize the model, I constructed a sequence S by repeating a portion of HXB2 reference sequence from position 2227 to 2292 so as match the predefined fixed length. The epitome was then initialized with a near uniform distribution but with sequence S as its mode. (Note that the repeated HXB2 sequence has a coverage of only 5%.) Upon convergence, the sequence corresponding to the mode of learned distribution was obtained. The resulting nucleotide sequence was scored for coverage of the 9-mers. As an example, Figure 6.8 illustrates parts of the learned multiframe epitome. The figure focuses on two frames in the epitome and illustrates how it captures some of the variability in the HXB2 region discussed in the introduction.

The coverage of the multiframe optimized string was reduced due to the need to disallow stop codons (which terminate translation of the protein) in frames which coded for 9-mers. It is possible to deal with this issue in numerous ways, such as introducing Lagrange multipliers to penalize objective function, or introducing additional data in the set consisting solely of stop codons followed by start codons, which may force a natural resolution to the problem. I chose to post-process sequences by inserting a start codon at the first triplet in the frame which was not used for coding in other frames.

The multiframe epitome provides the vaccine with the best coverage of the three techniques(see Table 6.1). We should note that vaccines consisting of random strains unnecessarily

Table 6.1: Coverage of all 9-mers, between HXB2 positions 2227 to 2292, from all viruses in the host population

Length (nucleotides)	Random strains	Single frame	Multiframe Epitome
180	20.45%	9.55%	20.65%
300	23.18%	22.27%	30.45%
600	41.36%	40.91%	57.1%

repeat conserved regions, but use multiple frames. Thus, for short vaccines, they outperform packing in the amino acid domain (single frame greedy optimization), but they lose ground to both optimization techniques for longer constructs.

I tested the method on a second dataset of real sequences originating from the Hepatitis B virus. I obtained 42 sequences from the Hepatitis Virus Database maintained at National Institute of Genetics, Japan. Hepatitis B also utilizes multiple frames for Pol and S genes. The S gene is wholly contained in the Pol gene region, with a frame shift. I extracted the first half of the overlapping region 315 nucleotides from each sequence and submitted this dataset to a greedy algorithm operating on single frame and an EM algorithm operating on multiple frames. I allowed the length of the vaccine to vary over 300, 600 and 900 nucleotides. The coverage results are given in Table 6.2.

Ideally, one would apply this method to sequences from regions where the virus utilizes multiple frames. I wanted to explore the sensitivity of the method to sequences which contain a mixture of multiple frame and single frame regions. In order to test the behavior of the algorithm, I synthesized a set of sequences which contains such a mix of regions. For a set of sequences, I first generated a prototype sequence, consisting of a multiframe and a single

Table 6.2: Coverage of Hepatitis B 9-mers extracted from positions 155 to 454 in the reference sequence. This region contains the S and Pol gene of Hepatitis B in two different frames .

Length	Greedy	EM
300	13.3%	23.3%
600	19.8%	32.9%
900	24.9%	35.7%

frame portion. For the multiframe portion, I randomly generated a sequence which encodes two different "genes" with a frame shift of 1. The single frame portion was appended to the multiframe portion as a continuation of the first frame. Hence, the first frame codes for a longer gene. From the prototypical sequence, I generated 10 sequences with a mutation probability of 1% for each nucleotide, simulating the high rate of mutation observed in viruses. In the multiframe portion of the sequence, I only accepted mutations which do not introduce stop codons in either frame. The single frame portion was mutated without regard to the second frame, hence the second frame may contain stop codons. In addition, I generated 4 different sets with varying length ratios of the multiframe vs single frame portions. I kept the length of the multiframe sequence fixed at 222 nucleotides and allowed the single frame portion length to vary over 111, 222, 442, 660. I refer to these different setups as 2:1, 1:1, 1:2 and 1:3 ratio scenarios.

I compared the performance of the algorithm to the performance of a greedy algorithm with a fixed vaccine length of 300 nucleotides. Note that over the different ratios 2:1, 1:1, 1:2 and 1:3, the length of the sequences grows, and the multiframe portion contributed less and less in terms of coverage.

The Table 6.3 illustrates coverage results for the two methods. The clear winner is the EM method, even in the extreme example where a single frame contains $\frac{4}{5}$ of the coding sequences. Hence, the method is robust to the presence of single coding frame regions.

Table 6.3: Coverage of all 9-mers in synthetic sequences composed of a multiframe and a single frame portion. The ratio of lengths of multiframe vs. single frame regions correspond to the columns.

Method	2:1	1:1	1:2	1:3
Greedy	26.69%	32.11%	20.95%	15.23%
Multiframe Epitome (EM)	50.75%	43.37%	35.41%	25.18%

6.5 Challenges to coverage optimization

In previous sections, I have addressed the task of vaccine design as coverage optimization, as well as approaches to incorporating models of immunogenicity and cross-reactivity. All of the peptides were treated as desirable and potential candidates for inclusion anywhere in the vaccine. However, a particular composition of peptides may impact efficacy of vaccine, via immunodominance effects, or may even be harmful to the human, by inducing autoimmune response. While there is strong evidence for these two effects, exact characterization of either of them is not yet available. The aim of this section is to suggest possible approaches to optimization methods which take into account these effects and leverage methods presented in the thesis.

6.5.1 Negative design

We may wish to preclude certain peptides from ever occurring in a vaccine under design. Examples of undesirable peptides are infrequent but immunodominant peptides, and epitopes which cross-react with “self” peptides. Even though negative selection during maturation of T-cells (see Section 2.4.1) eliminates T-cells which react to human peptides, some viral infections can activate T-cells that recognize self peptides and kill healthy cells. This kind of response is called an autoimmune response. Examples of autoimmune response induced by a

viral infection have been shown in model organisms [57] as well as humans [83]. The relationship between vaccines and autoimmune response is still being debated [74]. In the context of synthetic vaccines, additional care must be taken not to include peptides which may induce autoimmune response. Synthetic vaccines, even the simplest consensus sequence, may contain peptides which do not necessarily occur in the wild type virus. Hence the absence of autoimmune response following HIV infection need not guarantee an absence of autoimmune response following immunization. Another reason for excluding peptides from the vaccine is that these peptides may be "bad" epitopes in that they may lead to higher viral load [44].

In this section, I explore possible computational methods which would allow us to optimize vaccines for coverage while excluding undesirable peptides. For any given pair of peptides d^t and d^s in the dataset which in a particular overlap produce an undesirable peptide, we will introduce a prior $p^{ts}(i^t, i^s)$ on the two variables i^t and i^s. Recall these are the variables that indicate which position in the epitome was used to generate peptides d^t and d^s respectively. The idea is that this prior should assign low probability to configurations of i^t and i^s that may give rise to an undesirable peptide, while keeping the rest of the configurations equally likely.

Note that any particular composition of two peptides d^t and d^s can be represented by the difference of the two positions. For example, $i^t - i^s = L$ corresponds to placing peptide d^s right before peptide d^t, without any overlap between peptides. Let $U^{ts}(x)$ denote a function which evaluates to 0 if peptides d^t and d^s placed with an offset difference of x would produce an undesirable peptide, and 1 otherwise. The prior for a particular pair of peptides d^s and d^t is then given by composition $p^{ts}(i^t = i, i^s = j) \propto U^{ts}(i - j)$. For a pair of peptides d^t and d^s which can never be composed so as to give rise to an undesirable peptide the function U^{ts} will be constant and the prior will be uniform across all possible assignments of i^t and i^s.

A set of precluded peptides may give rise to a number of precluded compositions of different peptides. Many pairs of peptides may be assigned a nonuniform prior. We will denote a set of peptide indices which have a precluded composition with peptide d^t as J^t. Then for all $s \in J^t$ there is a nonuniform prior $p^{ts}(i^t, i^s)$. As a result, each pair of variables for which

we introduce a nonuniform prior will be coupled in inference. The cost of inference then becomes exponential in the size of the biggest set J^t, since the posterior on variable i^t depends on assignment to all of the variables i^s, where $s \in J^t$.

In order to alleviate this computational burden we can introduce a variational approximation of the posterior $q(i)$ in the form of $\prod_t q^t(i)$.

Updates for this approximate posterior are given by:

$$q^t(i^t = i) \propto \left(\prod_{s \in J^t} \epsilon^{\sum_j q^s(i^s=j)(1-U^{ts}(i-j))} \right) \log p(d^t|i^t, E)p(i^t)$$

where ϵ denotes a small constant corresponding to $\log 0$, and is used in order to avoid numerical issues.

Note that empirically none of the peptides in HIV or the vaccines generated for Nef, Gag or Pol contained any of the peptides which occur in human proteins. However, a relatively dissimilar peptide in the vaccine may indeed cross react with a self peptide.

6.5.2 Immunodominance

Immunodominance refers to the effect of a particular epitope suppressing reaction to a different epitope. The impact of this effect on vaccine design lies in its ability to limit the number of epitopes that the immune system can learn to recognize from a vaccine. One explanation of immunodominance posits a predator-prey model between infected cells and T-cells [67]. Consider an infected cell exhibiting many different epitopes as prey. An efficient predator, a T-cell recognizing a particular epitope, may swiftly eliminate most of the infected cells; this prevents other, less efficient T-cells from examining the infected cells and other epitopes. Even with equally efficient T-cells, an imbalance may be introduced by differences in efficiency of processing and presentation of different epitopes. One possible approach to dealing with immunodominance is to subdivide the vaccine into multiple components. This diversifies the prey, the infected cells, which may be targeted by different T-cells. Further improvement of this approach may be possible by using predictive models of processing and presentation. A

set of peptides can be stratified according to the efficiency of their presentation, or their experimentally observed immunogenicity. A set of components can be constructed out of each of the strata, which will have comparable presentation efficiency or immunogenicity. Regardless of the subdivision of the peptide sets across different vaccine components, the assembly algorithms presented in this thesis may be used to construct each of the components.

Chapter 7

Conclusion

A vaccine aiming to provide protection to a human population against a viral population should take into account the diversity of both of these populations. This is especially important for viruses such as HIV, because a complete viral population cannot be represented by a single virus; rather, we must choose how to represent the population in the vaccine so as to offer the human population the best protection possible. The vaccine should offer each of the recipients sufficient protection against most of the infecting viruses. In the case of a vaccine targeting CD8 T-cell responses, protection is afforded to a recipient if his immune system can learn from the vaccine a set of epitopes which occur in the bulk of HIV viruses. In this thesis, I formulate an objective function, the optimum of which should coincide with such a vaccine. This objective function requires a vaccine to include as many of the frequent peptides in the vaccine as possible. The guiding principle is that excluding a potential peptide from a consideration for inclusion in the vaccine can be done only if there are strong indications that a peptide is not in fact an epitope for some members of the human population receiving the vaccine. On the other hand, an infrequent epitope need not be included in the vaccine, since even if it is a potent epitope, its infrequency in the viral population makes it of little use when protecting against most of the infecting viruses.

This is the first computational framework aimed squarely at vaccine design. The framework specifies an immunologically motivated objective function, the coverage function. After showing that optimization of the objective function is NP-hard, I introduce an exact algorithm and, as the NP-hardness of the problem justifies the use of approximations, several approximate methods for optimizing the objective function. The exact optimization is based on branch-and-cut methodology. The simplest of the approximate methods is the greedy algorithm, related to techniques for solving the shortest superstring problem. I also develop probabilistic methods based on an epitome model. The optimal solution for the vaccine design problem is shown to be one of the global maxima of the likelihood of the model. An EM algorithm is derived which is used to obtain maximum likelihood estimates, from which a vaccine candidate can be obtained. A number of extensions of this basic EM algorithm are derived under assumptions about parametrization of the model and the approximate posterior probability of peptide placement in the epitome. In addition, I introduce a model which allows for simultaneously separating out low importance peptides and building a vaccine. All of these methods can produce sequences of variable length, allowing utilization of a range of capacities of the vaccine delivery vectors. I have also generalized the existing methods for vaccine design, center-of-tree and consensus, to methods which can be used to build vaccines longer than a single HIV virus, in order to allow comparison with the techniques which optimize coverage.

All of these coverage optimization methods– greedy optimization, exact branch-and-cut technique and EM algorithm based methods –are assessed in-silico in terms of their objective function scores. The clear winner, the exact method, is not feasible to apply on large datasets. This leaves the epitome algorithm, a close second best in terms of coverage, but substantially faster than the branch-and-cut method with its exponential run time. Furthermore, these methods perform better on coverage optimization than the existing methods for vaccine design: center-of-tree, consensus and their generalizations.

In-vitro validation based on ELISPOT assay indicates that coverage optimized vaccines may offer protection in a large number of infections, up to 89%, compared to the 50.50%

achievable by consensus vaccines. However, the assays give an approximation of how an immune system which takes full advantage of the vaccine may respond to an incoming infection. This level of protection may be hampered by effects such as immunodominance, or improved by broad cross-reactivity of included peptides. Further validation is performed under an assumption of the effects of immunodominance imposing a strict limit on the number of epitopes which can be learned from a vaccine. With a modest maximal number of 6 epitopes, the epitome vaccine still achieves a mean protection of 65.83%, compared to the mean consensus protection of 41.73%.

I also show that further reduction in the required length of the vaccine can be achieved if the compression inherent in HIV viruses, multiple frame coding, is utilized. The possible challenges to synthetic vaccines based on coverage optimization in both safety and feasibility remain as areas of future research, for example assessing the effects cross-reactivity and countering immunodominance. Modeling cross-reactivity effectively is incredibly important for vaccine design, as it will provide better estimates of the true coverage of the viral population. Employing human genome data to generate examples of peptides which most likely do not cross-react with viral peptides may allow construction of cross-reactivity models. Addressing the issue of immunodominance in vaccine design by suitable sequence design is also of tremendous interest since systematic analysis of the actual mechanisms of immunodominance have not even been attempted, hence constructing a model of this process is not likely at this time. However, in the same way that, to a certain extent, coverage obviates the need for epitope prediction, an intelligent vaccine construction technique may obviate need for an immunodominance model.

In a practical scenario where a vaccine designer aims at finding a sequence with high coverage on a new dataset, I recommend running the best performing EM technique, with a large number of random restarts. The greedy technique can also be run so as to provide a reasonable solution quickly. If sufficient time and computing resources are available, running the branch-and-cut method may provide a near optimal solution, as it stores the current best solution.

The coverage optimization framework, along with the derived algorithms is immunologically valid both in theory and as supported by in-vitro experiments. Moreover, it is easily extendable, as shown by the variety of approximate probabilistic methods derived. Chapter 6 points the way towards more sophisticated vaccine design in the coverage optimization framework, illustrating its potency and relevance for future developments in the field.

Bibliography

[1] Tamir A. An o(pn2) algorithm for the p-median and related problems on tree graphs. *Operations Research Letters*, 19:59–64(6), August 1996.

[2] Todd M Allen, Xu G Yu, Elizabeth T Kalife, Laura L Reyor, Mathias Lichterfeld, Mina John, Michael Cheng, Rachel L Allgaier, Stanley Mui, Nicole Frahm, Galit Alter, Nancy V Brown, Mary N Johnston, Eric S Rosenberg, Simon A Mallal, Christian Brander, Bruce D Walker, and Marcus Altfeld. De novo generation of escape variant-specific CD8+ T-cell responses following cytotoxic T-lymphocyte escape in chronic human immunodeficiency virus type 1 infection. *J Virol*, 79(20):12952–12960, Oct 2005.

[3] J P Anderson, A G Rodrigo, G H Learn, Y Wang, H Weinstock, M L Kalish, K E Robbins, L Hood, and J I Mullins. Substitution model of sequence evolution for the human immunodeficiency virus type 1 subtype B gp120 gene over the C2-V5 region. *J Mol Evol*, 53(1):55–62, Jul 2001.

[4] R M Anderson, J Swinton, and G P Garnett. Potential impact of low efficacy HIV-1 vaccines in populations with high rates of infection. *Proc Biol Sci*, 261(1361):147–151, Aug 1995.

[5] R Andino, D Silvera, SD Suggett, PL Achacoso, CJ Miller, D Baltimore, and MB Feinberg. Engineering poliovirus as a vaccine vector for the expression of diverse antigens. *Science*, 265(5177):1448–1451, 1994.

[6] Manoj Bhasin and G P S Raghava. Analysis and prediction of affinity of TAP binding peptides using cascade SVM. *Protein Sci*, 13(3):596–607, Mar 2004.

[7] Manoj Bhasin and G P S Raghava. Pcleavage: an SVM based method for prediction of constitutive proteasome and immunoproteasome cleavage sites in antigenic sequences. *Nucleic Acids Res*, 33(Web Server issue):202–207, Jul 2005. Evaluation Studies.

[8] Andreas Bråve, Karl Ljungberg, Britta Wahren, and Margaret A Liu. Vaccine delivery methods using viral vectors. *Molecular pharmaceutics*, 4(1), 2007.

[9] Christian Brander, Nicole Frahm, and Bruce D Walker. The challenges of host and viral diversity in HIV vaccine design. *Curr Opin Immunol*, 18(4):430–437, Aug 2006.

[10] Dennis R Burton, Ronald C Desrosiers, Robert W Doms, Mark B Feinberg, Robert C Gallo, Beatrice Hahn, James A Hoxie, Eric Hunter, Bette Korber, Alan Landay, Michael M Lederman, Judy Lieberman, Joseph M McCune, John P Moore, Neal Nathanson, Louis Picker, Douglas Richman, Charles Rinaldo, Mario Stevenson, David I Watkins, Steven M Wolinsky, and Jerome A Zack. Public health. A sound rationale needed for phase III HIV-1 vaccine trials. *Science*, 303(5656):316, Jan 2004.

[11] P Cascio, C Hilton, A F Kisselev, K L Rock, and A L Goldberg. 26S proteasomes and immunoproteasomes produce mainly N-extended versions of an antigenic peptide. *EMBO J*, 20(10):2357–2366, May 2001.

[12] J M Coffin. HIV population dynamics in vivo: implications for genetic variation, pathogenesis, and therapy. *Science (New York, N.Y.)*, 267(5197).

[13] Jon Cohen. Public health. AIDS vaccine still alive as booster after second failure in Thailand. *Science*, 302(5649):1309–1310, Nov 2003. News.

[14] Thomas M. Cover and Joy A. Thomas. *Elements of Information Theory*. John Wiley & Sons, Inc., N. Y., 1991.

[15] K A Crandall. *The evolution of HIV*. Johns Hopkins University Press. Baltimore, MD, 1999.

[16] Anne S De Groot, Luisa Marcon, Elizabeth A Bishop, Daniel Rivera, Michele Kutzler, David B Weiner, and William Martin. HIV vaccine development by computer assisted design: the GAIA vaccine. *Vaccine*, 23(17-18):2136–2148, Mar 2005.

[17] L Deml, A Bojak, S Steck, M Graf, J Wild, R Schirmbeck, H Wolf, and R Wagner. Multiple effects of codon usage optimization on expression and immunogenicity of DNA candidate vaccines encoding the human immunodeficiency virus type 1 Gag protein. *J Virol*, 75(22):10991–11001, Nov 2001.

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

[19] K Deres, H Schild, K H Wiesmuller, G Jung, and H G Rammensee. In vivo priming of virus-specific cytotoxic T lymphocytes with synthetic lipopeptide vaccine. *Nature*, 342(6249):561–564, Nov 1989.

[20] N A Doria-Rose, G H Learn, A G Rodrigo, D C Nickle, F Li, M Mahalanabis, M T Hensel, S McLaughlin, P F Edmonson, D Montefiori, S W Barnett, N L Haigwood, and J I Mullins. Human immunodeficiency virus type 1 subtype B ancestral envelope protein is functional and elicits neutralizing antibodies in rabbits similar to those elicited by a circulating subtype B envelope. *J Virol*, 79(17):11214–11224, Sep 2005. Comparative Study.

[21] Rika Draenert, Marcus Altfeld, Christian Brander, Nesli Basgoz, Colleen Corcoran, Alysse G Wurcel, David R Stone, Spyros A Kalams, Alicja Trocha, Marylyn M Addo, Philip J R Goulder, and Bruce D Walker. Comparison of overlapping peptide sets for de-

tection of antiviral CD8 and CD4 T cell responses. *J Immunol Methods*, 275(1-2):19–29, Apr 2003. Comparative Study.

[22] N P Emmerich, A K Nussbaum, S Stevanovic, M Priemer, R E Toes, H G Rammensee, and H Schild. The human 26 S and 20 S proteasomes generate overlapping but different sets of peptide fragments from a model protein substrate. *J Biol Chem*, 275(28):21140–21148, Jul 2000.

[23] J Felsenstein. Evolutionary trees from DNA sequences: a maximum likelihood approach. *J Mol Evol*, 17(6):368–376, 1981.

[24] J. Felsenstein. *Inferring Phylogenies*. Sinauer Associates, Sunderland, MA, 2003.

[25] M Fischetti, J J S Gonzalez, and P Toth. Solving the orienteering problem through branch-and-cut. *INFORMS Journal on Computing*, 10:133–148, 1998.

[26] Thomas C Friedrich, Elizabeth J Dodds, Levi J Yant, Lara Vojnov, Richard Rudersdorf, Candice Cullen, David T Evans, Ronald C Desrosiers, Bianca R Mothe, John Sidney, Alessandro Sette, Kevin Kunstman, Steven Wolinsky, Michael Piatak, Jeffrey Lifson, Austin L Hughes, Nancy Wilson, David H O'Connor, and David I Watkins. Reversion of CTL escape-variant immunodeficiency viruses in vivo. *Nat Med*, 10(3):275–281, Mar 2004.

[27] Alan M. Frieze and Wojciech Szpankowski. Greedy algorithms for the shortest common superstring that are asymptotically optimal. *Algorithmica*, 21(1):21–36, 1998.

[28] N Froloff, A Windemuth, and B Honig. On the calculation of binding free energies using continuum methods: application to MHC class I protein-peptide interactions. *Protein Sci*, 6(6):1293–1301, Jun 1997.

[29] J Gallant, D Maier, , and J Storer. On finding minimal length superstrings. *Journal of Computer and System Sciences*, 20:50–58, 1980.

[30] Feng Gao, Bette T Korber, Eric Weaver, Hua-Xin Liao, Beatrice H Hahn, and Barton F Haynes. Centralized immunogens as a vaccine strategy to overcome HIV-1 diversity. *Expert Rev Vaccines*, 3(4 Suppl):161–168, Aug 2004.

[31] Feng Gao, Eric A Weaver, Zhongjing Lu, Yingying Li, Hua-Xin Liao, Benjiang Ma, S Munir Alam, Richard M Scearce, Laura L Sutherland, Jae-Sung Yu, Julie M Decker, George M Shaw, David C Montefiori, Bette T Korber, Beatrice H Hahn, and Barton F Haynes. Antigenicity and immunogenicity of a synthetic human immunodeficiency virus type 1 group M consensus envelope glycoprotein. *J Virol*, 79(2):1154–1163, Jan 2005.

[32] D N Garboczi, P Ghosh, U Utz, Q R Fan, W E Biddison, and D C Wiley. Structure of the complex between human T-cell receptor, viral peptide and HLA-A2. *Nature*, 384(6605):134–141, Nov 1996. Comment.

[33] Brian Gaschen, Jesse Taylor, Karina Yusim, Brian Foley, Feng Gao, Dorothy Lang, Vladimir Novitsky, Barton Haynes, Beatrice H Hahn, Tanmoy Bhattacharya, and Bette Korber. Diversity considerations in HIV-1 vaccine selection. *Science*, 296(5577):2354–2360, Jun 2002.

[34] Marc P Girard, Saladin K Osmanov, and Marie Paule Kieny. A review of vaccine research and development: the human immunodeficiency virus (HIV). *Vaccine*, 24(19):4062–4081, May 2006.

[35] B G Golden, L Levy, and R Vohra. The orienteering problem. *Naval Res. Logist*, 34:307–318, 1991.

[36] S. Louis Hakimi. Optimum location of switching centers and the absolute centers and medians of a graph. *Operations Research*, 12:450–459, 1964.

[37] David Heckerman, Carl Myers Kadie, and Jennifer Listgarten. Leveraging information across HLA alleles/supertypes improves epitope prediction. In *RECOMB*, pages 296–308, 2006.

[38] IAVI database of AIDS vaccines in human trials. http://www.iavireport.org/trialsdb, Jun 2006.

[39] N Jojic, B J Frey, and A Kannan. Epitomic analysis of appearance and shape. *International Conference on Computer Vision*, 01:34, 2003.

[40] Nebojsa Jojic, Vladimir Jojic, Brendan J. Frey, Christopher Meek, and David Heckerman. Using epitomes to model genetic diversity: Rational design of hiv vaccines. In *NIPS*, 2005.

[41] Nebojsa Jojic, Manuel Reyes-Gomez, David Heckerman, Carl Kadie, and Ora Schueler-Furman. Learning MHC I - peptide binding. In *ISMB (Supplement of Bioinformatics)*, pages 227–235, 2006.

[42] Can Kesmir, Alexander K Nussbaum, Hansjorg Schild, Vincent Detours, and Soren Brunak. Prediction of proteasome cleavage motifs by neural networks. *Protein Eng*, 15(4):287–296, Apr 2002.

[43] Girish S Kesturu, Bonnie A Colleton, Yi Liu, Laura Heath, Obaid Shakil Shaikh, Charles R Jr Rinaldo, and Raj Shankarappa. Minimization of genetic distances by the consensus, ancestral, and center-of-tree (COT) sequences for HIV-1 variants within an infected individual and the design of reagents to test immune reactivity. *Virology*, 348(2):437–448, May 2006.

[44] Photini Kiepiela, Kholiswa Ngumbela, Christina Thobakgale, Dhanwanthie Ramduth, Isobella Honeyborne, Eshia Moodley, Shabashini Reddy, Chantal de Pierres, Zenele Mncube, Nompumelelo Mkhwanazi, Karen Bishop, Mary van der Stok, Kriebashnie Nair, Nasreen Khan, Hayley Crawford, Rebecca Payne, Alasdair Leslie, Julia Prado, Andrew Prendergast, John Frater, Noel McCarthy, Christian Brander, Gerald H Learn, David Nickle, Christine Rousseau, Hoosen Coovadia, James I Mullins, David Hecker-

man, Bruce D Walker, and Philip Goulder. CD8+ T-cell responses to different HIV proteins have discordant associations with viral load. *Nat Med*, 13(1):46–53, Jan 2007.

[45] A F Kisselev, T N Akopian, K M Woo, and A L Goldberg. The sizes of peptides generated from protein by mammalian 26 and 20 S proteasomes. Implications for understanding the degradative mechanism and antigen presentation. *J Biol Chem*, 274(6):3363–3371, Feb 1999.

[46] B T Korber, B F Foley, C l Kuiken, S K Pillai, , and J G Sodroski. Numbering positions in HIV relative to HXB2CG. In Korber et al., editor, *Human Retroviruses and AIDS*. Los Alamos National Laboratory, 1998.

[47] Bette T. M. Korber, Christian Brander, Barton F. Haynes, Richard Koup, John P. Moore, Bruce D. Walker, , and David I. Watkins. HIV Molecular Immunology 2005. Technical report, Los Alamos National Laboratory, Theoretical Biology and Biophysics,Los Alamos, New Mexico., 2005. LA-UR 06-0036.

[48] A Lalvani, R Brookes, S Hambleton, W J Britton, A V Hill, and A J McMichael. Rapid effector function in CD8+ memory T cells. *The Journal of experimental medicine*, 186(6).

[49] Mette Voldby Larsen, Claus Lundegaard, Kasper Lamberth, Soren Buus, Soren Brunak, Ole Lund, and Morten Nielsen. An integrative approach to CTL epitope prediction: a combined algorithm integrating MHC class I binding, TAP transport efficiency, and proteasomal cleavage predictions. *Eur J Immunol*, 35(8):2295–2303, Aug 2005.

[50] P Liljeström and H Garoff. A new generation of animal cell expression vectors based on the semliki forest virus replicon. *Bio/technology (Nature Publishing Company)*, 9(12), 1991.

[51] Jinyan Liu, Bonnie A Ewald, Diana M Lynch, Anjali Nanda, Shawn M Sumida, and Dan H Barouch. Modulation of DNA vaccine-elicited CD8+ T-lymphocyte epitope im-

munodominance hierarchies. *J Virol*, 80(24):11991–11997, Dec 2006. Comparative Study.

[52] Shan Lu. Combination DNA plus protein HIV vaccines. *Springer seminars in immunopathology*, 28(3), 2006.

[53] J. MacQueen. Some methods for classification and analysis of multivariate observations. *Proceedings of the Fifth Berkeley Symposium on Mathematics, Statistics and Probability*, 1:281–296, 1967.

[54] S G E Marsh, E D Albert, W F Bodmer, R E Bontrop, B Dupont, H A Erlich, D E Geraghty, J A Hansen, C K Hurley, B Mach, W R Mayr, P Parham, E W Petersdorf, T Sasazuki, G M Th Schreuder, J L Strominger, A Svejgaard, P I Terasaki, and J Trowsdale. Nomenclature for factors of the HLA system, 2004. *Tissue antigens*, 65(4).

[55] M Mata, P J Travers, Q Liu, F R Frankel, and Y Paterson. The MHC class I-restricted immune response to HIV-gag in BALB/c mice selects a single epitope that does not have a predictable MHC-binding motif and binds to Kd through interactions between a glutamine at P3 and pocket D. *J Immunol*, 161(6):2985–2993, Sep 1998.

[56] Iglesias MC, Mollier K, Beignon AS, Souque P, Adotevi O, Lemonnier F, and Charneau P. Lentiviral Vectors Encoding HIV-1 Polyepitopes Induce Broad CTL Responses In Vivo. *Mol Ther*, Mar 2007. JOURNAL ARTICLE.

[57] Lori McCoy, Ikuo Tsunoda, and Robert S Fujinami. Multiple sclerosis and virus induced immune responses: autoimmunity can be primed by molecular mimicry and augmented by bystander activation. *Autoimmunity*, 39(1).

[58] F E McCutchan. Understanding the genetic diversity of HIV-1. *AIDS (London, England)*, 14 Suppl 3, 2000.

[59] Olivier Michielin and Martin Karplus. Binding free energy differences in a TCR-peptide-MHC complex induced by a peptide mutation: a simulation analysis. *J Mol Biol*, 324(3):547–569, Nov 2002.

[60] C B Moore, M John, I R James, F T Christiansen, C S Witt, and S A Mallal. Evidence of HIV-1 adaptation to HLA-restricted immune responses at a population level. *Science*, 296(5572):1439–1443, May 2002.

[61] M G Morgado, M L Guimaraes, and B Galvao-Castro. HIV-1 polymorphism: a challenge for vaccine development - a review. *Mem Inst Oswaldo Cruz*, 97(2):143–150, Mar 2002.

[62] James I Mullins, David C Nickle, Laura Heath, Allen G Rodrigo, and Gerald H Learn. Immunogen sequence: the fourth tier of AIDS vaccine design. *Expert Rev Vaccines*, 3(4 Suppl):151–159, Aug 2004.

[63] R M Neal and G E Hinton. A view of the EM algorithm that justifies incremental, sparse, and other variants. pages 355–368, 1999.

[64] David C Nickle, Mark A Jensen, Geoffrey S Gottlieb, Daniel Shriner, Gerald H Learn, Allen G Rodrigo, and James I Mullins. Consensus and ancestral state HIV vaccines. *Science*, 299(5612):1515–1518, Mar 2003. Comment.

[65] J P Nkolola, E G-T Wee, E-J Im, C P Jewell, N Chen, X-N Xu, A J McMichael, and T Hanke. Engineering RENTA, a DNA prime-MVA boost HIV vaccine tailored for Eastern and Central Africa. *Gene Ther*, 11(13):1068–1080, Jul 2004.

[66] V Novitsky, H Cao, N Rybak, P Gilbert, M F McLane, S Gaolekwe, T Peter, I Thior, T Ndung'u, R Marlink, T H Lee, and M Essex. Magnitude and frequency of cytotoxic T-lymphocyte responses: identification of immunodominant regions of human immunodeficiency virus type 1 subtype C. *J Virol*, 76(20):10155–10168, Oct 2002.

[67] M A Nowak, R M May, and K Sigmund. Immune responses against multiple epitopes. *Journal of theoretical biology*, 175(3).

[68] A K Nussbaum, T P Dick, W Keilholz, M Schirle, S Stevanovic, K Dietz, W Heinemeyer, M Groll, D H Wolf, R Huber, H G Rammensee, and H Schild. Cleavage motifs of the yeast 20S proteasome beta subunits deduced from digests of enolase 1. *Proc Natl Acad Sci U S A*, 95(21):12504–12509, Oct 1998.

[69] L Patterson, B Peng, X Nan, and M. Robert-Guroff. Live adenovirus recombinants as vaccine vectors. In Levine M., Kaper J., Rappuoli R., Liu M., and Good M., editors, *In New Generation Vaccines 3rd ed*, pages 325–35. Marcel Dekker: New York, 2004.

[70] A S Perelson, A U Neumann, M Markowitz, J M Leonard, and D D Ho. HIV-1 dynamics in vivo: virion clearance rate, infected cell life-span, and viral generation time. *Science (New York, N.Y.)*, 271(5255).

[71] Carlo Federico Perno, Valentina Svicher, and Francesca Ceccherini-Silberstein. Novel drug resistance mutations in HIV: recognition and clinical relevance. *AIDS Rev*, 8(4):179–190, Oct 2006.

[72] H Rammensee, J Bachmann, N P Emmerich, O A Bachor, and S Stevanovic. SYFPEITHI: database for MHC ligands and peptide motifs. *Immunogenetics*, 50(3-4):213–219, Nov 1999.

[73] Steven E Raper, Marc Yudkoff, Narendra Chirmule, Guang-Ping Gao, Fred Nunes, Ziv J Haskal, Emma E Furth, Kathleen J Propert, Michael B Robinson, Susan Magosin, Heather Simoes, Lisa Speicher, Joseph Hughes, John Tazelaar, Nelson A Wivel, James M Wilson, and Mark L Batshaw. A pilot study of in vivo liver-directed gene transfer with an adenoviral vector in partial ornithine transcarbamylase deficiency. *Hum Gene Ther*, 13(1):163–175, Jan 2002.

[74] Ami Schattner. Consequence or coincidence? the occurrence, pathogenesis and significance of autoimmune manifestations after viral vaccines. *Vaccine*, 23(30).

[75] Mahender Singh. No vaccine against HIV yet–are we not perfectly equipped? *Virol J*, 3:60, 2006.

[76] G L Smith and B Moss. Infectious poxvirus vectors have capacity for at least 25 000 base pairs of foreign dna. *Gene*, 25(1), 1983.

[77] Z Sweedyk. A 2 1/2-approximation algorithm for shortest superstring. *SIAM J. Comput.*, 29(3):954–986, 1999.

[78] Frédéric Tangy and Hussein Y Naim. Live attenuated measles vaccine as a potential multivalent pediatric vaccination vector. *Viral immunology*, 18(2), 2005.

[79] Jojic V, Jojic N, Heckerman D, Kadie K, Meek C, Moore C, John M, , and Mallal S. Hla-driven optimization of an hiv vaccine immunogen. In *Proceedings of the 12th Conference on Retroviruses and Opportunistic Infections*, 2005.

[80] J A Wolff, R W Malone, P Williams, W Chong, G Acsadi, A Jani, and P L Felgner. Direct gene transfer into mouse muscle in vivo. *Science*, 247(4949 Pt 1):1465–8, 1990.

[81] L A Wolsey. *Integer Programming*. John Wiley & Sons, 1998.

[82] Guang Lan Zhang, Nikolai Petrovsky, Chee Keong Kwoh, J Thomas August, and Vladimir Brusic. PREDTAP: a system for prediction of peptide binding to the human transporter associated with antigen processing. *Immunome Res*, 2:3, 2006.

[83] Z S Zhao, F Granucci, L Yeh, P A Schaffer, and H Cantor. Molecular mimicry by herpes simplex virus-type 1: autoimmune disease after viral infection. *Science*, 279(5355):1344–7, 1998.