Advanced Immunoinformatics Approaches for Precision Medicine

Dissertation

der Mathematisch-Naturwissenschaftlichen Fakultät
der Eberhard Karls Universität Tübingen
zur Erlangung des Grades eines
Doktors der Naturwissenschaften
(Dr. rer. nat.)

vorgelegt von

M.Sc. Benjamin Schubert

aus Heidelberg

Tübingen 2016

Gedruckt mit Genehmigung der Mathematisch-Naturwissenschaftlichen Fakultät der Eberhard Karls Universität Tübingen.

Tag der mündlichen Qualifikation:
25.04.2017

Dekan:
Prof. Dr. Wolfgang Rosenstiel

1. Berichterstatter:
Prof. Dr. Oliver Kohlbacher

2. Berichterstatter:
Prof. Dr. Can Keşmir

"It is more important to know what sort of person has a disease, than to know what sort of disease a person has."

Hippocrates of Cos (c. 460 BC - c. 370 BC)

Abstract

Genomic sequencing and other '-omic' technologies are slowly changing biomedical practice. As a result, patients now can be treated based on their molecular profile. Especially the immune system's variability, in particular that of the human leukocyte antigen (HLA) gene cluster, makes such a paradigm indispensable when treating illnesses such as cancer, autoimmune diseases, or infectious diseases. It can be, however, costly and time-consuming to determine the HLA genotype with traditional means, as these methods do not utilize often pre-existing sequencing data. We therefore proposed an algorithmic approach that can use these data sources to infer the HLA genotype. HLA genotyping inference can be cast into a set covering problem under special biological constraints and can be solved efficiently via integer linear programming. Our proposed approach outperformed previously published methods and remains one of the most accurate methods to date.

We then introduced two applications in which a HLA-based stratification is vital for the efficacy of the treatment and the reduction of its adverse effects. In the first example, we dealt with the optimal design of string-of-beads vaccines (SOB). We developed a mathematical model that maximizes the efficacy of such vaccines while minimizing their side effects based on a given HLA distribution. Comparisons of our optimally designed SOB with experimentally tested designs yielded promising results. In the second example, we considered the problem of anti-drug antibody (ADA) formation of biotherapeutics caused by HLA presented peptides. We combined a new statistical model for mutation effect prediction together with a quantitative measure of immunogenicity to formulate an optimization problem that finds alterations to reduce the risk of ADA formation. To efficiently solve this bi-objective problem, we developed a distributed solver that is up to 25-times faster than state-of-the art solvers. We used our approach to design the C2 domain of factor VIII, which is linked to ADA formation in hemophilia A. Our experimental evaluations of the proposed designs are encouraging and demonstrate the prospects of our approach.

Bioinformatics is an integral part of modern biomedical research. The translation of advanced methods into clinical use is often complicated. To ease the translation, we developed a programming library for computational immunology and used it to implement a Galaxy-based web server for vaccine design and a KNIME extension for desktop PCs. These platforms allow researchers to develop their own immunoinformatics workflows utilizing the platform's graphical programming capabilities.

Zusammenfassung

Genomics und andere '-omics' Technologien verändern langsam die biomedizinische Praxis. Sie erlauben die molekulare Charakterisierung von Patienten und damit eine individualisierte Abstimmung der Therapie. Gerade die Variabilität des Immunsystems - insbesondere im Bereich der Human Leukocyte Antigen (HLA) Gruppe - hat einen starken Einfluss auf die Effektivität einer Therapien und muss daher bei der Entwicklung von neuen Behandlungsmethoden von Krebs-, Infektions- und Autoimmunerkrankungen beachtet werden. Die herkömmlichen Methoden zur Identifikation eines HLA Genotyps können jedoch zeit- und kostenintensiv sein. In vielen Kliniken werden allerdings bereits standardmäßig Sequenzdaten für andere diagnostische Zwecke erhoben. Um diese vorhandenen Daten auch für die HLA-Genotypisierung nutzbar zu machen, wird im ersten Teil dieser Arbeit ein algorithmisches Verfahren vorgestellt, dass in der Lage ist, den HLA Genotyp eines Patienten akkurat abzuleiten. Das Problem wurde als Mengenüberdeckungsproblems formuliert und konnte in einem Leistungsvergleich alle bereits publizierten Methoden übertreffen.

Im weiteren Verlauf dieser Arbeit wurden zwei Anwendungsbeispiele vorgestellt bei denen eine HLA-basierte Stratifizierung nötig ist um die Effektivität der Therapie zu gewährleisten und um deren Nebenwirkungen zu reduzieren. Im ersten Beispiel wurde ein neues Modell zur Polypeptid-Vakzin Entwicklung vorgestellt. Das mathematische Modell optimiert die Wirksamkeit des Impfstoffes und reduziert dessen Nebenwirkungen ausgehend von einer gegebenen HLA-Verteilung. Vergleiche von optimierten Polypeptidvakzinen mit experimentell getesteten Konstruktionen lieferten vielversprechende Ergebnisse. Im zweiten Beispiel wurde das Problem der Anti-Arzneimittel-Antikörper (engl., Anti-Drug-Antibody, ADA) Bildung behandelt. Hierzu wurde ein algorithmisches Verfahren vorgestellt, das ein neuartiges statistisches Modell zur Abschätzung von Mutationseffekten mit einem quantitativen Maß für Immunogenität kombiniert um Modifikationen des Biotherapeutikums zu identifizieren, die das Risiko der ADA-Formation senken. Um das bikriterielle Optimierungsproblem effizient zu lösen, wurde ein Lösungsverfahren entwickelt, das 25-fach schneller ist als Standardtechniken. Das vorgestellte Verfahren wurde genutzt, um die C2 Domäne von Faktor VIII, die für die ADA formation in Hemophilia A Patienten veranwortlich ist, zu modifizieren. Die anschließende experimentelle Evaluation der berechneten Modifikationen war ermutigend und demonstrierte das Potential der vorgestellten Methode.

Die Bioinformatik ist ein integraler Bestandteil der modernen biomedizinischen Forschung. Die Translation von computergestützten Methoden in die klinische Anwendung ist allerdings oft kompliziert. Um diesen Schritt zu erleichtern, wurden mehrere Softwarelösungen entwickelt, die im zweiten Teil dieser Arbeit näher beschrieben werden. Zunächst wurde eine Programmierschnittstelle für Immunoinformatik konzipiert, die im dann genutzt wurde um einen Galaxy-basierten Webservers für Vakzindesign und eine KNIME-Erweiterung für Immunoinformatik zu entwickeln. Diese beiden Plattformen ermöglichen es Forschern, ihre eigenen Immunoinformatik-Workflows mit Hilfe der grafischen Programmierumgebungen beider Softwarelösungen zu entwerfen.

Acknowledgments

First of all, I would like to thank my supervisor Prof. Oliver Kohlbacher for giving me the opportunity and freedom to explore my own ideas, while always providing me with guidance when I asked for it. Thank you for your constant support and the many discussions we had.

I also would like to thank my reviewers Prof. Can Keşmir for agreeing to invest the time and effort to review this work.

Furthermore, I am grateful to all my wonderful colleagues and students I had the pleasure of working with. I enjoyed our coffee breaks, discussions, and collegial atmosphere. A special thanks goes to the former and present members of the computation immunology SIG: first of all Nora Toussaint, who introduced me to this interesting field, Magdalena Feldhahn, Pierre Dönnes, Andras Szolek, Christopher Mohr, Linus Backert, and Mathias Walzer. I also would like to thank my collaborators: Charlotta Schärfe, Luis de la Garza, Thomas Hopf, and Debora Marks.

A special thanks goes to Luis for taking care of the coffee machine. Without you, the whole department would fall apart!

I am grateful for my friends and family, especially Fabian, Björn, Andreas, Charlotta, and Simon for our semi-frequent gaming nights.

My deepest gratitude goes to Tabea Rosenkranz, who constantly reminded me that there is a life outside of work, and patiently endured me during the last months of writing.

In accordance with the standard scientific protocol, I will use the personal pronoun ***we*** to indicate the reader and the writer, or my scientific collaborators and myself.

Contents

Chapter 1

Introduction

The ever decreasing costs of genome sequencing and other '-omic' technologies allows the characterization of an individual's molecular state at an unprecedented level. This wealth of data is slowly changing how patients are treated. Diseases can now be categorized based on their molecular aberrations, patients can be treated based on their molecular characteristics, and new drugs can be developed that take these molecular variations into account. This paradigm shift in biomedical research and its application is summarized under the umbrella term *precision medicine* (PM). PM attempts to tailor a customized treatment of an individual's disease based on the gathered patient- as well as disease-related molecular, lifestyle and environmental information to increase the efficacy of the therapy while decreasing side-effects. It also allows targeting diseases that could not be treated effectively before due to their highly personalized characteristics, such as cancer and rare genetic diseases.

The molecular characterization of diseases allows the development of highly effective drugs that target specific molecular aberrations, but only work in a particular subpopulation of patients. Vemurafenib for example is successfully used to treat melanoma but is only effective in tumors exhibiting specific BRAF mutations (V600E, V600K)[1]. Patients which do not possess these tumor mutations, however, do not benefit; the drug might even promote tumor growth[2,3]. Molecular screening also allows the repurposing of already existing drugs for other diseases in so called 'off label' approaches. In our example, the same BRAF variation is also observed in pulmonary adenocarcinoma, even though less frequently. Consequently, a study has shown that patients respond to Vemurafenib in such cases as well[4]. Especially patients with rare diseases benefit the most from molecular screening approaches. A study conducted by Zhu *et al.* demonstrated the usage of genomic analysis which resulted in the diagnosis of 24% of prior undiagnosed patients[5]. Such analyses could eventually lead to a therapy that would otherwise be impossible to identify.

Ultimately, the goal of PM is to transform our healthcare system from a reactive to a preventive system by recognizing the onsets of a disease much earlier and by treating each patient with the best, most personalized treatment possible.

The Role of the Immune System in Precision Medicine

The immune system, primarily adaptive immunity, plays a significant role in many disease etiologies including that of cancer, autoimmune diseases, and infectious diseases. It is, therefore, necessary to account for the genetic variability of the immune system when making a treatment decision or while developing new therapies. Especially the *human leukocyte antigen* (HLA) cluster of genes is the source of the largest part of the adaptive immune system's variability. In fact, it is the most polymorphic region of the human genome. Due to its high polymorphism and polygenicity, it is very unlikely that any two individuals express the same HLA genotype. The polymorphism is also the reason

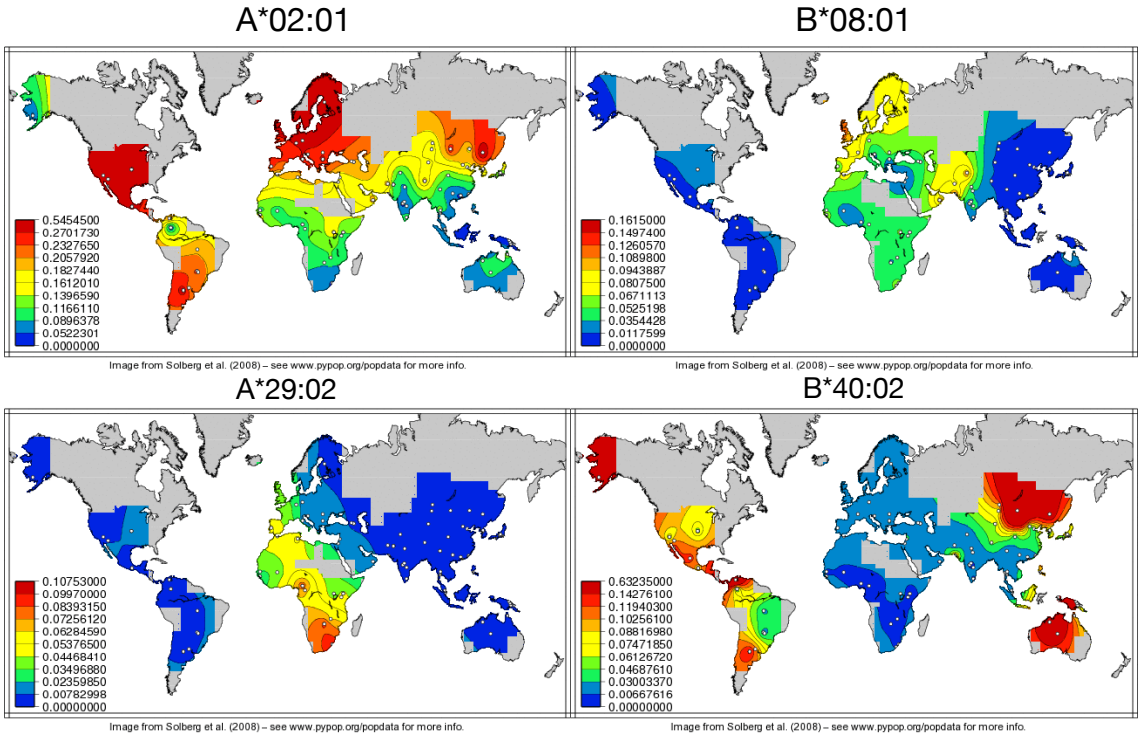


Figure 1.1: Worldwide distribution of four selected HLA alleles. Images by Solberg *et al.*[6]

why the HLA allele distribution varies drastically between populations and geographic regions (Figure 1.1). The HLA genes encode membrane-bound molecules that present small peptides to other cells of the immune system. Upon binding of the HLA-peptide (pHLA) complex, immune cells, in particular T-lymphocytes, are activated which leads to

an immune response and ultimately to the elimination of the source protein (called *antigen*) of the peptide. These proteins can be either of pathogenic origin or aberrations of human proteins such as tumor antigens. Each HLA molecule exhibits a slightly different specificity towards subsets of peptides. Thus, the genetic variability of HLA, among others factors, shape the space of peptides the immune system can act on. It is therefore indispensable to stratify patients based on their HLA genotype for immunomodulatory treatments and to account for the HLA allele prevalence in a population when developing a new vaccines and other therapeutic products.

As of now, the identification of a patient's HLA genotype is labor-intensive and uses complex protocols that require the generation of biological data solely for the purpose of identifying the HLA genotype. Many clinical centers, however, have started to routinely sequence their patient's genomes for diagnostic and therapeutic purposes. Thus, our first scientific contribution presented in this work is the development of an algorithmic and mathematical solution to identify a patient's HLA genotype utilizing these routinely generated data. The method casts the HLA typing problem into a well-studied combinatorial optimization problem that can be efficiently solved via *integer linear programming*. With this algorithmic approach, the diagnostic phase of treatment can be accelerated, costs reduced, and extensive sequencing studies like the 1,000 Genomes Project[7] or the Cancer Genome Atlas project[8] can be analyzed retrospectively.

Individual HLA genotyping has seen great use in oncology and especially in individualized cancer vaccine immunotherapy. Here, the unique genetic variation of tumor cells is exploited to find altered peptide sequences that uniquely characterize the tumor. These peptides, so-called *neo-epitopes*, can be used as therapeutic vaccines to redirect the patient's immune system on the cancerous cells again. The patient's HLA genotype has to be characterized first, however, to identify neo-epitopes that can bind to the HLA molecules of the patient via algorithmic or experimental means. Once a suitable set of neo-epitopes is selected, the peptides have to be assembled into a vaccine. One particular approach that combines the neo-epitopes by concatenating them like beads on a string has been widely and successfully used. These so-called *string-of-beads* vaccines are either administered as *synthetic long peptides* (SLP)[9,10] or as RNA/DNA minigene constructs[11,12]. The efficacy of these string-of-beads vaccines relies heavily on the correct processing within the cells. Especially proteasomal and additional cleavage events have a substantial impact on vaccine efficacy. If the string-of-beads is incorrectly cleaved, therapeutic neo-epitopes are destroyed and additional artificial peptides can arise with unwanted immune reactions[13]. Amino acids adjacent to the cleavage site, which are determined by the ordering of the neo-epitopes, influence these cleavage events[14]. To positively modulate the peptide cleavage and thus increase vaccine efficacy, several groups started using short spacer sequences to connect two neo-epitopes[13,15,16]. But, as of now, no systematic approach exists to determine the optimal

ordering and spacer sequence for a particular string-of-beads vaccine. We therefore developed a mathematical framework that can determine the optimal spacer sequence and length in conjunction with the optimal neo-epitope ordering to increase the recovery likelihood of each neo-epitope while decreasing the probability of unwanted immune reactions.

The need of HLA genotype-based stratification has also become apparent in other areas of drug development. For many therapeutically used proteins, inefficacy and side-effects have been linked to immunological effects. These side-effects stem from non-self peptides that are presented on HLA molecules and originate from the biotherapeutic. Presentation and recognition of these peptides (called epitopes) lead to the activation of B cells that consequently produce antibodies specifically targeting the biotherapeutic. The antibodies bind the biotherapeutic, which leads to its neutralization, thus reducing or even nullifying the intended therapeutic effect. In severe cases, especially when the biotherapeutic is used as replacement therapy, these anti-drug antibody (ADA) formations can lead to systemic, potentially life-threatening, autoimmune reactions[17]. To decrease the risk of ADA formation, several groups have started targeting specifically the presented epitopes of a particular biotherapeutic by introducing single point alterations to hinder the epitope binding to HLA molecules[18–20]. The introduction of amino acid alterations can potentially negatively influence the structural stability and function of the biotherapeutic. Thus, it is necessary to find alterations that (1) hinder the ADA causing epitopes to bind to HLA and (2) do not disrupt the structural and functional integrity of the biotherapeutic at the same time. Experimentally identifying such mutations is extraordinarily time-consuming and resource-intensive. The process usually entails immunogenicity testing of overlapping peptides of the biotherapeutic, alanine screenings or other mutagenesis techniques on the identified immunogenic regions to find immunogenicity reducing mutations, incorporating those mutations into the full protein sequence, and finally testing for functionality and overall immunogenicity[21]. To redirect experimental efforts to only promising designs, hereby reducing time and cost expenses, we developed an optimization framework that can efficiently deal with the inherent multi-objectivity of the design problem. It finds designs that maximally reduce the immunogenicity of the biotherapeutic while minimally disrupting its structural and functional integrity. The optimization framework can be used in a stratified manner taking the HLA distribution of particular population into account, or, in a very personalized setting, considering only the HLA genotype of a patient during optimization.

The Role of Bioinformatics in Precision Medicine

Precision medicine heavily relies on new technologies and computational models to identify the specific patient and disease characteristics to make informed decisions on how to treat

a patient optimally and design new drugs based on this information. Hence, bioinformatics is an integral part of precision medicine, dealing with genetic data storage, development of reliable and reproducible analysis workflows, and the creation of mathematical models to support the development of new personalized therapies.

The development time of such workflows is often long due to non-standardized data formats and software interfaces of bioinformatics tools. While some fields of bioinformatics, for example parts of the genomics and proteomics community, have developed their own standards, other research areas such as immunoinformatics have not yet experienced such consolidation. Especially the lack of a unified software interface and output formats of epitope prediction methods makes interoperability difficult. We therefore developed a Python module called *FRamework for Epitope Discovery* (FRED) 2 that acts as a unifying layer between state-of-the-art immunoinformatics tools and the workflow developer to enable interoperability and provides identical output formats and many pre- and post-processing functionalities routinely used in such applications. Overall, it decreases development time and allows for rapid prototyping.

Even with the use of FRED 2, the development of immunology related workflows is still a challenging task and requires trained software engineers or bioinformaticians. To translate advanced immunoinformatics methods into a daily working environment of practitioners, clinicians, and biologists, two main problems have to be overcome: (1) Bioinformatics software is often very complex and challenging to install; (2) the required software solution should have a simple user interface, but must be powerful enough to allow the user to combine single components interactively to complex workflows. Web-based solutions like Galaxy[22] circumvent the problem of installing and configuring software packages and enable the user to connect individual components to workflows with a simple graphical user interface via drag-and-drop. We thus developed a Galaxy-based web service specifically targeting immunoinformatics-relevant applications such as HLA genotyping, epitope prediction, and vaccine design.

Often data volume or legal restrictions such as data protection and privacy agreements in biomedical research restrict the use of web services. Thus, local desktop solutions are needed that still retain the benefits of a Galaxy-based web service. One possible software platform that possesses such qualities is the Konstanz Information Miner (KNIME)[23,24]. KNIME is a desktop data analytics and reporting platform that enables the user to design workflows graphically out of individual components, called nodes. Via Generic KNIME nodes (GKN)[25], any command line tool can be easily integrated into KNIME. To avoid the necessity of installing the required bioinformatics tools that might run only on some operation systems, we extended GKN to interact seamlessly with so-called containerized applications. Docker is a flat virtual environment that allows for convenient packaging of pre-configured software and to execute them on any operating system. Together with

the extended GKN, we developed an immunoinformatics application toolbox mimicking the functionality of the Galaxy web service in conjunction with a Docker container that comprises of all necessary software applications.

Thesis Outline

The thesis is structured in eight chapters. First, the necessary immunological and methodological background in Chapters 2 and 3 is introduced respectively. Then the mathematical model for HLA genotyping is derived and evaluated in Chapter 4, followed by the description and evaluation of the string-of-beads design framework in Chapter 5. Chapter 6 describes the mathematical model an algorithm to de-immunize biotherapeutics, and Chapter 7 introduces the developed software applications. Finally, Chapter 8 concludes this thesis and provides an outlook to future research questions.

Chapter 2

Biological Background

This chapter establishes the biological background of this thesis by providing a general overview of the immune system in Section 2.1 followed by a detailed description of the cellular and humoral immune response in Sections 2.3 and 2.4 respectively. For a more comprehensive introduction, the reader is kindly referred to Kindt *et al.*[26] or Murphy *et al.*[27].

2.1 The Immune System

The main function of the immune system is to detect and eliminate infectious agents (pathogens) and abnormal cells (e.g., cancer cells) within an organism. The immune system must thus be able to distinguish between various pathogens like viruses, bacteria, and parasites, or abnormal cell from own, healthy cells. In vertebrates, two distinct, but intertwined, defense systems evolved: the *innate* and the *adaptive immune* system. The innate immune system acts as the first line of defense and includes physical barriers like the skin, mucus membranes, and physiological mechanisms like increasing body temperature. It is also capable of recognizing a broad array of pathogen-associated molecule patterns (PAMPs) with generic pattern recognition receptors (PRRs), which lead to inflammatory responses, and the recruitment of other immune cells.

The moment the infection reaches a critical stage, the immune system activates its second line of defense, the adaptive immune system (Section 2.2), via attracted *antigen presenting cells* (APCs). These APCs are capable of phagocytosing infectious agents for degradation (Section 2.3.2). Once an APC has phagocytized a pathogen, it migrates towards lymph nodes to present degraded pathogenic peptides, so-called *epitopes*, to naïve cells of the adaptive immune system. These cells transform to effector cells, which are able to induce an individually tailored immune response against the invading pathogen.

2.2 Adaptive Immunity

The *adaptive immune system* is capable of adapting and learning from its encounter with a pathogen. Once a pathogen has been recognized, cells of the adaptive immune system develop into *memory cells* that act stronger and more rapidly upon a repeated encounter of the same pathogen. These memory cells determine the *immunological memory* and guarantee a long-term protection or *immunity* against a pathogen.

The major cellular components of the adaptive immune system are T and B cells, which belong to the family of lymphocytes. These cells carry receptors capable of recognizing foreign pathogenic proteins, so-called *antigens*. An antigen is not recognized in its entirety; only a small peptide sequence, called *epitope*, interacts with the T- and B-cell receptors. On recognition, the B and T cells proliferate into short-lived effector and long-lived memory cells. The effector cells actively eliminated the recognized antigen, while the memory cells form a long lasting protection against the same antigen.

The T-cell antigen receptor (TCR) is a membrane-bound receptor that is only capable of recognizing epitopes once they are bound to a family of membrane proteins encoded by genes of the *major histocompatibility complex* (MHC, Section 2.3.1). In humans this gene cluster is also known as *human leukocyte antigen* (HLA). The induced adaptive immune reaction of a T cell is called *cellular immune response* and will be outlined in Section 2.3.

The antigen receptor of B-cells (BCR) is a membrane-bound, Y-shaped protein called *antibody*. In contrast to TCRs, antibodies can also recognize epitopes of antigens that are (free-floating) in solution. Similar to dendritic cells, B cells are also capable of presenting epitopes to T cells and are therefore classified as professional APCs. The adaptive immune reaction induced by B-cell recognition is called *humoral immune response* and will be described in Section 2.4.

2.3 Cellular Immune Response

The main protagonists of the cellular immune response are T cells, which originate in the bone marrow and mature in the thymus. During maturation, the TCR variant is formed using a genetic recombination mechanism that results in over a billion possible sequence variants. The TCR is a heterodimer with each chain consisting of both constant and variable regions. Rearrangement of genes that encode the different segments of light and heavy chain generates the vast sequence diversity. Cells that have not produced a correctly folded TCR are driven into apoptosis; the remaining cells undergo positive and negative selection in the thymus. The positive selection ensures that T cells can adequately bind to HLA molecules; weakly binding T cells go into apoptosis. At this stage, the T-cell subtype is also determined, depending on the HLA molecule class they first interact with. The

negative selection on the other hand guarantees that no T cells mature that are capable of binding self-epitopes and thus would induce an autoimmune reaction. T cells that successfully underwent the positive and negative selection, the so-called *naïve T cells* are released into the blood, where they circulate between the blood stream and the lymph nodes patrolling for APCs presenting epitopes they can recognize.

T cells can be classified into three subtypes according to their primary functionality and by the co-stimulation receptors they express: T helper (T_H), cytotoxic (CTL), and regulatory T cells (T_{REG}). T_H cells additionally express the cluster of differentiation 4 receptor (CD4) on their surface that stabilizes the interaction of the TCR with HLA-II molecules.

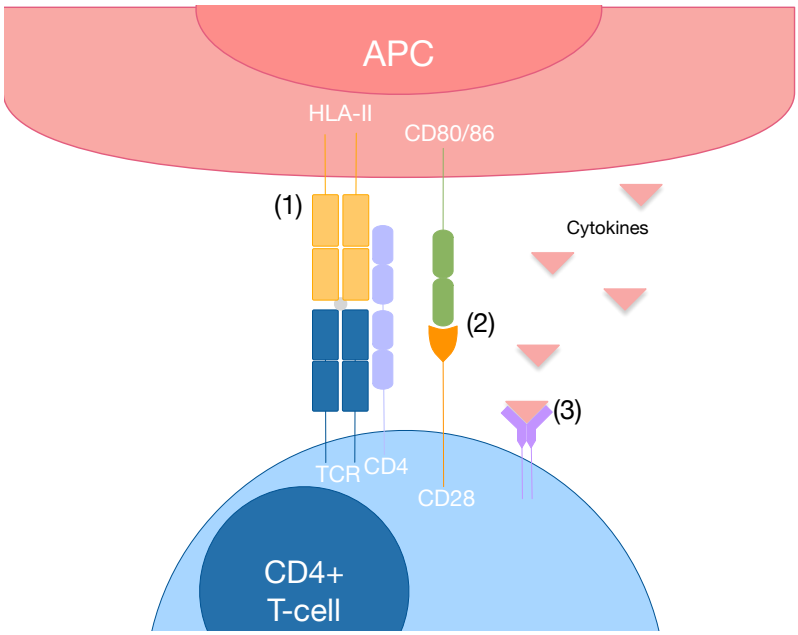


Figure 2.1: The three singals in T-cell activation. The interaction between epitope-HLA and T-cell receptor (1) together with its co-stimulatory signal caused by the interaction between CD28 and CD80/86 activates the T cell. This leads to the secretion of cytokines that stimulate the T cell to proliferate and affect the differentiation into subtype clones.

CTLs express the co-receptor CD8. CTLs kill infected cells or abnormal cells by recognizing non-self epitopes bound to HLA-I. CD8, similar to CD4, interacts with the exterior of HLA-I molecules and stabilizes the complex. Once an antigen is identified as non-self, perforins and granzymes are released forming pores in the infected cell's membrane causing the lysis of the target cell. Released granzymes additionally penetrate the infected cell through the formed pores and induce the apoptosis of the cell.

T_{REG} regulate the immune response and suppress other T effector cells preventing excessive or autoimmune reactions. The suppressive effect is induced by direct cell-to-cell

contact, secretion of immune suppressive cytokines, and deprivation of immune stimulating cytokines. T_{REG} are characterized by the expression of CD4, FoxP3, and CD25 and are most likely derived from naïve T helper cells[28].

The activation of a T cell is controlled by three major signals (Figure 2.1). The TCR binding of the epitope-HLA complex (pHLA) strengthened by the T cell's co-receptors CD4/8, is the primary activating signal. The second signal is given by the interaction of another T-cell co-receptor, CD28, and its ligands CD80/86 on the APC. Without a co-stimulation via CD28, the T cell will go into an inactivated state, called *anergy*, which is another mechanism to prevent autoimmune reactions against proteins that are not expressed in the thymus during T cell maturation. Both signals are required for the activation of a naïve T cell, which then drastically changes the expression of various surface proteins. The activation also triggers a signal cascade that eventually leads to the increased production of interleukin 2 (IL-2). IL-2 and other cytokines constitute the third signal that is primarily responsible for the differentiation of the T cells into their respective subtypes. Once a naïve T cell has proliferated into effector and memory cells, the second and third signal is not needed to activate the cell to perform its primary duties.

2.3.1 The Major Histocompatibility Complex

HLA molecules play an important role in the adaptive immune system and fall into one of two classes, HLA-I and HLA-II. HLA-I molecules are expressed on all human nucleated cells, whereas HLA-II molecules are exclusively expressed on APCs. HLA-I molecules

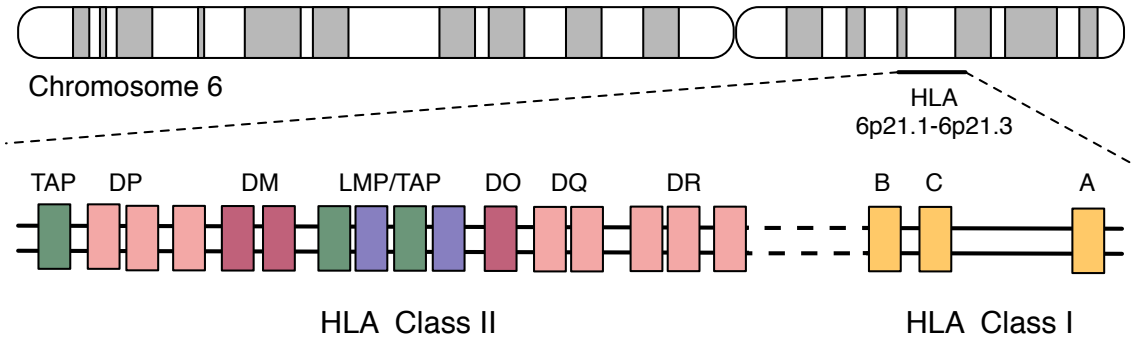


Figure 2.2: The human leukocyte antigen (HLA) gene cluster. The HLA gene cluster is located at the small arm of chromosome 6. It comprises genes that encode for the HLA class I and II molecules, besides other genes involved in the immune system. HLA-I and HLA-II are polygenic and encode for three genes respectively - A, B, and C in case of HLA-I, and DR, DQ, and DP in case of HLA-II.

consist of two α-subunits and a smaller β_2 microglobulin stabilizing the complex. The α_1 and α_2 domains form the enclosed epitope-binding groove, restricting the HLA-I epitope length to 8-11 amino acids (AA), which are intracellularly derived. HLA-II molecules, on

the other hand, consist of an α- and a β chain. The binding groove of HLA-II molecules is formed by α_1 and β_1 subunits, is open on both ends, and presents extracellular epitopes of length 13 to 18 amino acids. But only a peptide substring of about nine amino acids directly interacts with the HLA-II binding groove and the TCR[29].

Since HLA molecules are responsible for the presentation of pathogenic peptides, a large evolutionary pressure is exerted upon pathogens to escape the presentation by HLA. The polygenesis and polymorphism of HLA impede immune evasion to some extent. Both molecules are encoded by multiple genes, which are co-dominantly expressed. The HLA-I α-chain is encoded by three major (HLA-A, -B, and -C) and three minor (HLA-E, -F, and -G) genes, whereas the α- and β-chain of HLA-II are encoded by three major (HLA-DR, -DP, -DQ) and two minor (HLA-DO, and -DM). HLA-DO and -DM are used during antigen processing to load and stabilize the HLA-II molecule and are not expressed on the surface of APCs. The HLA genes cluster is located on chromosome 6p21 and is $\sim$ 3 mega base pairs (bp) long (Figure 2.2). HLA-I genes are composed of eight approximately 275 bp long exons, of which the most polymorphic exons 2 and 3 encode for the binding groove. HLA-DRA gene, encoding for the DR α, is composed of five exons. The β-chain encoding DRB genes consist of six exons, of which only exon 3 contains all polymorphisms relevant for epitope binding. Both genes encoding for the α- and β-chain of HLA-DQ and HLA-DP, consist of five exons, and carry polymorphisms relevant for epitope binding. Genetic linkages between and within the HLA-II genes increase the genetic complexity. Strong linkage disequilibrium exists between HLA-DR and HLA-DQ alleles. Also, a varying number of HLA-DRB genes are expressed in different combinations[30]. Although nine HLA-DRB loci have been identified, only a few are present in distinct combinations in every individual.

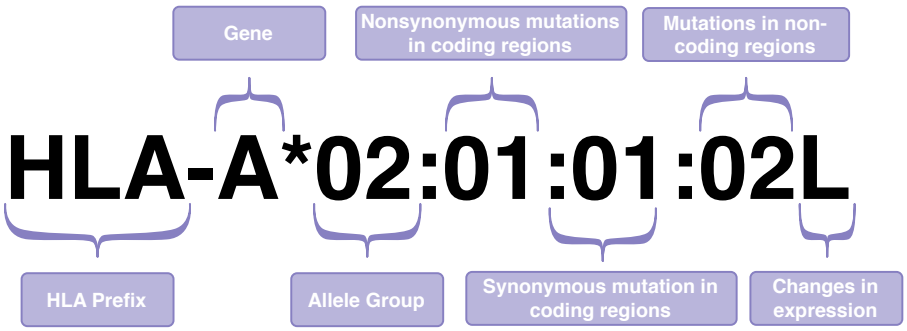


Figure 2.3: The official HLA allele nomenclature proposed by the WHO Nomenclature Committee for Factors of the HLA System[31].

Today, more than 10,574 HLA-I and 3,658 HLA-II alleles are known that encode for 7,563 HLA-I molecules and 4,094 HLA-II molecules (IEDB[32], release 3.23.0). Due to the polygenicity, codominant expression and high polymorphism, each human expresses three to six different HLA-I and up to eight HLA-II different molecules.

To name each allele, a complex nomenclature was established[31](Figure 2.3). The first element describes the HLA gene separated by an asterisk followed by two digits indicating the locus. The next two digits describe the particular HLA protein, followed by synonymous variations within exons, and mutation in intronic regions. An alphabetic suffix is used to indicate the expression status of the HLA gene.

2.3.2 Antigen Processing

Antigens undergo a process called *antigen processing* that leads to the degradation of the proteins and subsequently to the presentation of antigenic peptides on HLA molecules (Figure 2.4). Depending on the origin of the antigens they go through slightly different processing steps.

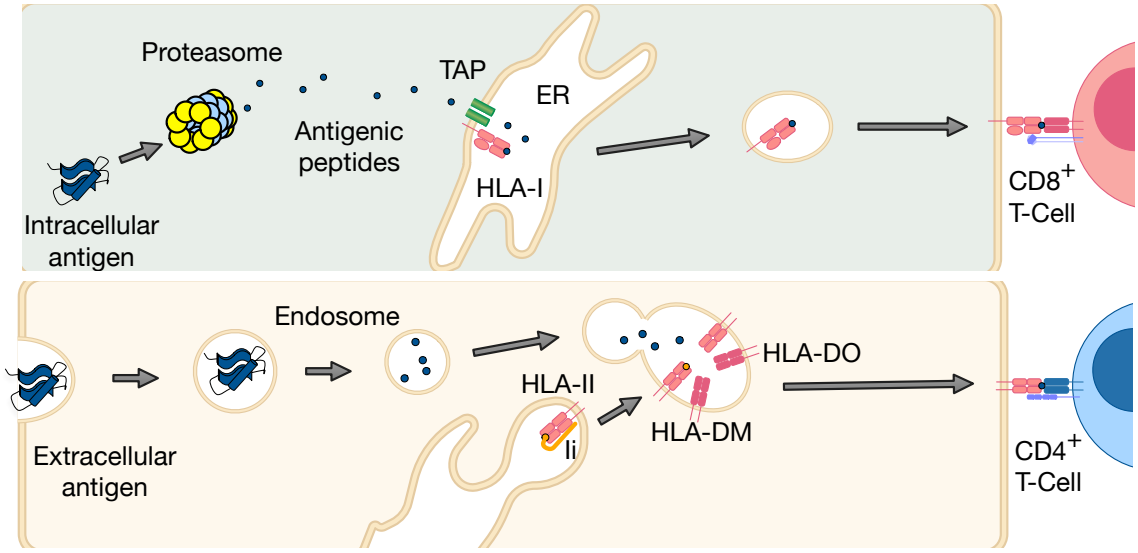


Figure 2.4: Top: The intracellular antigen processing pathway. Intracellular proteins of potential pathogenic origin are degraded by the proteasome and transported via TAP into the ER where the peptides are loaded onto HLA-I molecules. Bottom: The extracellular antigen-processing pathway. Extracellular antigens are phagocytized and degraded in endosomes by various proteases. The endosomal vesicles coalesce with lysosomes carrying HLA-II molecules. The peptide contained in the endosomes supplant CLIP, a small protein fragment that protected the peptide-binding groove from loading intracellular peptides during HLA class II folding and transport from the ER. The loaded HLA molecules are transported to the surface of the cell where they present the loaded peptides to T lymphozytes.

Intracellular antigens, like any other intracellular protein, are eventually ubiquitinated and degraded by the proteasome, a protein complex equipped with several proteolytic subunits. The catalytic core of the proteasome is composed of four heptameric rings, forming a tube that is called the 20S proteasome[33]. While the outer rings consist of seven α subunits regulating the access, the inner rings contain seven β subunits, of which

β1, β2, and β5 are catalytically active. β1 cleaves after acidic residues, while β2 has a preference to cleave after basic residues, and β5 cleaves after hydrophobic residues[34]. Two regulator particles, called 19S, bind to one or both ends of the 20S proteasome, forming the so-called 26S proteasome. The 19S molecules facilitate the degradation of the ubiquitin chains and catalyze the unfolding of the antigen[33]. During immune response, stimulated by interferon α, β, and γ, the catalytic subunits are replaced by counterparts forming the so-called *immunoproteasome*. These subunits have an increased ability to cleave after basic and hydrophobic residues, which generates peptides that are more likely to bind to HLA-I molecules. The peptides produced by the proteasome are then transported into the *endoplasmatic reticulum* (ER) via the adenosine triphosphate-dependent *transporter associated with antigen processing* (TAP). In the ER, peptides are further degraded by *aminopeptidase associated with antigen processing* (ERAAP). The peptide-HLA-I complex is then transported to the cell surface via the Golgi apparatus.

Extracellular antigens are endo- or phagocytosed by APCs. The so formed double-layered endosome fuses with a lysosome, which leads to acidification of the compartment. Mediated by a decreasing pH, different proteases - cathepsins B, D, S, and L, as well as thiolreductase - are activated and degrade the antigens. The fused lysosome contains HLA-II molecules that were also folded in the ER. To protect the HLA-II molecule for loading intracellular peptides, its binding groove is blocked by the invariant chain (Ii). The invariant chain is degraded in the lysosome compartment, and a small fragment called *Class II-associated invariant chain peptide* (CLIP) remains in the binding groove to stabilize the HLA-II molecule[35]. CLIP is eventually replaced by a higher affinity extracellular peptide. HLA-DM supports the dissociation, fine-tuned by HLA-DO shaping the antigenic peptide repertoire[36]. The pHLA-II complex is exocytosed and presented on the APC's surface to CD4+ T cells.

2.4 Humoral Immune Response

Many bacteria and viruses linger in the extracellular space, either to proliferate or to migrate from cell to cell. The humoral immune response protects the extracellular space via antibody-secreting B cells. B cells develop from haematopoietic stem cells within the bone marrow. Similar to T cells, they undergo a positive and negative selection during development. The positive selection ensures that BCRs and even pre-BCRs are properly expressed on the surface. The negative selection recognizes autoreactive B cells that bind self-antigens expressed in the bone marrow milieu[37]. This leads to either *clonal deletion*, *receptor editing* via genetic recombination, or anergy. Once the selection process is finished, the immature B cells travel to the spleen and other secondary lymphoid tissues, where they are activated (Section 2.4.1). Upon activation, B cells differentiate into plasma cells,

which have a short lifespan and immediately secrete larger amounts of antibodies, and long living memory cells that allow a rapid immune response at the next encounter of the same antigen.

Antibodies bind to a specific antigen and reveal the pathogen's presence to specialized phagocytic cells (*opsonization*), or prevent the bacterial and viral adherence to healthy cells by blocking essential pathogenic membrane proteins (*neutralization*). A third mechanism of the antibody response activates the *complement system*, a set of plasma proteins that belong to the innate immune system. The complement enhances the opsonization and lysis of bacteria. To generate the needed diversity of 3×10^{11} distinct antibodies, a genetic recombination mechanism, similar to that of T-cell receptors, is used.

2.4.1 B-cell Activation

B-cell activation occurs in the secondary lymph nodes by binding either free-floating or antigens presented by APCs. In contrast to T cells, a B cell can recognize a wide range of antigen types, including polysaccharides, glycoproteins, lipopolysaccharides, and proteins. B cells are either activated in the presence of CD4+ T-cells (T-cell dependent activation, TD), or can be independently activated via extensive crosslinking of BCRs with the same antigen (T-cell independent activation, TI). TI activated B cell proliferate and form short-lived plasma cells, but tend to generate low affine, mostly IgM antibodies against the recognized antigen.

TD-activated B cells interact directly with CD4+ T cells by presenting epitopes of the BCR-bound antigen on HLA-II molecules. Interestingly, the HLA presented epitope is not necessarily required to be the same epitope that is recognized by the BCR, which opens up possibilities to modulate the B-cell response via T-cell epitope editing (Chapter 6). B cells also need a co-stimulatory signal, which originates from the interplay between the co-receptor CD40 expressed on the B-cell surface and the CD40L ligand of the T cell. The interacting T cell becomes polarized and secretes IL-4 and other cytokines in the direction of interaction. Together, these three signals stimulate the B cell to proliferate into short-lived plasma cells also secreting mostly IgM antibodies (Figure 2.5). But some of the activated B-cells form a *germinal center* (GC). Within this specialized microenvironment, B cells extensively proliferate (*clonal expansion*), and undergo *somatic hypermutation* to increase the antigen binding affinity by introducing point mutations. During the phase of clonal expansion, activated B cells are called *centroblasts* and express BCRs at low abundance. After a considerable number of proliferation cycles, the centroblasts move into another region of the GC, which is more enriched by CD4+ T cells, and dendritic cells. Here, the centroblast clones are called *centrocytes* and express a high number of BCRs on the surface. The centrocytes undergo a positive and negative selection process to ensure

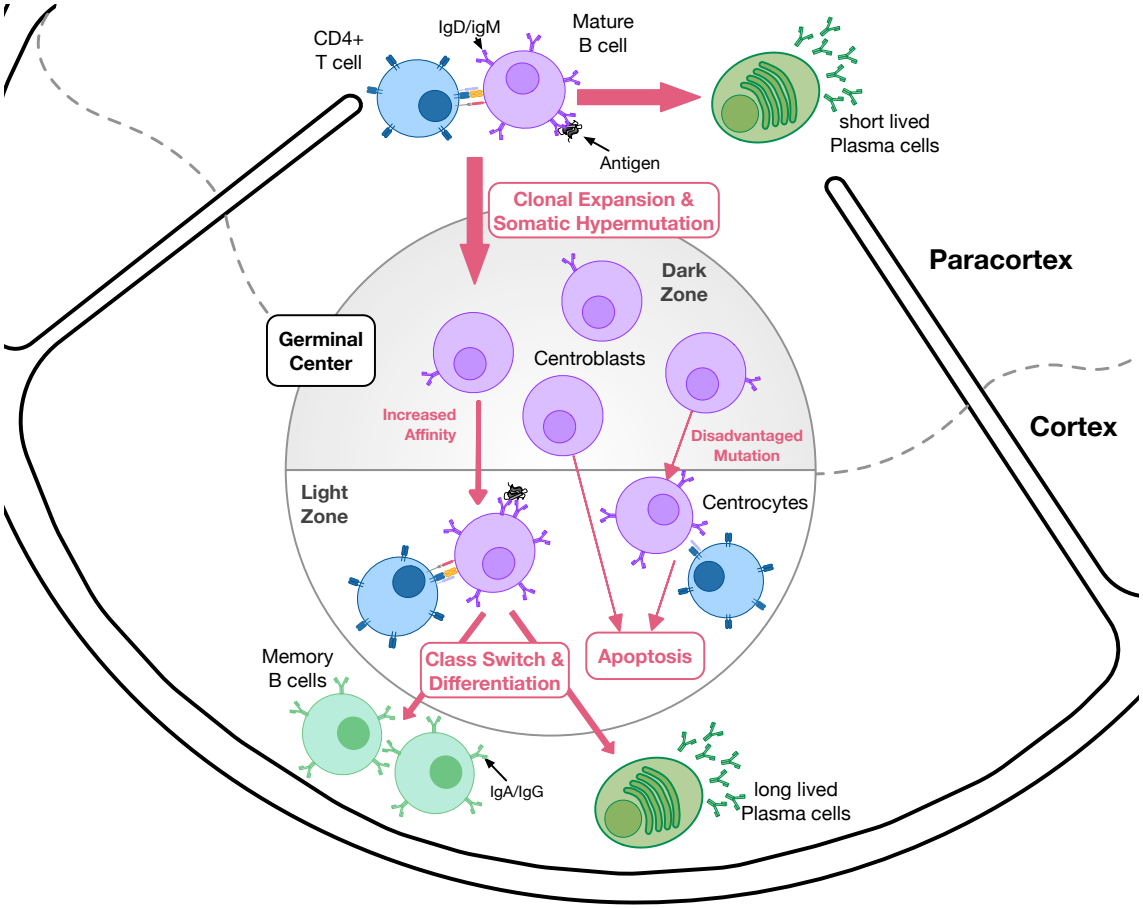


Figure 2.5: T cell-dependent activation of B cells. Upon interaction with CD4+ T cells that recognize epitopes derived from the B cell bound antigen, B cells are activated and form a germinal center. There, B cells undergo clonal expansion and somatic hypermutation. After positive and negative selection, the B-cell clones experience isotype switching and differentiate into different Ig subtypes.

that the hypermutated BCRs are still functioning and express an increased affinity for the foreign antigen. B cells that pass the GC undergo *isotype switching* and differentiate into memory cells or long-lived plasma cells secreting highly specific antibodies.

Chapter 3

Algorithmic Background

This chapter serves as an introduction to integer linear programming (Section 3.1) and multiobjective optimization (Section 3.2). It establishes the necessary theoretical concepts and algorithms to solve (multi-)objective combinatorial optimization problems. The reader is kindly referred to Wolsey *et al.*[38] and Ehrgott[39] for a comprehensive introduction to combinatorial optimization and multiobjective optimization.

3.1 Combinatorial Optimization

Combinatorial optimization is concerned with finding an optimal object minimizing or maximizing an *objective function* $z(\cdot)$, while imposing *constraints* on a finite set of possible alternatives. The combinatorial objects are represented by *variables* x_i, with $i \in \{1, .., n\}$, that are subject of the optimization. Their *domain* is a subset of the positive natural numbers $\mathbb{Z}_+^n$. Usually, these combinatorial objects have a concise interpretation (e.g., a particular permutation, or a path through a graph), and grow exponentially in the size of their variable domain, thereby prohibiting an exhaustive enumeration of all possible solutions in large combinatorial problems. In fact many combinatorial problems are known to be NP-hard (e.g., traveling salesman problem, knapsack problem, boolean satisfiability problem), but there exist also special cases of hard problems and combinatorial problems for which polynomial time algorithms exist (e.g., minimum spanning tree, shortest path, sequence alignment). Depending on the form of the objective function and constraints, optimization problems can be categorized into different classes. In this thesis, only linear objective functions and constraints are considered, which define the class of *integer linear programs* (ILPs). Many combinatorial problems can be expressed as an ILP, which is generally formulated as follows:

$$\begin{aligned} &\min \mathbf{c}^T\mathbf{x} \\ &\text{s.t. } \mathbf{A}\mathbf{x} \geq \mathbf{b} \\ &\mathbf{x} \in \mathbb{Z}_+^n, \end{aligned} \tag{3.1}$$

where $\mathbf{c} \in \mathbb{R}^n$ is a constant coefficient vector, and $\mathbf{A} \in \mathbb{R}^{n \times m}$ is a constant coefficient matrix. $\mathbf{b} \in \mathbb{R}^m$ is the constant vector representing the lower bounds of the constraints. The set $\mathcal{X} := \{\mathbf{x} \in \mathbb{Z}_+^n | \mathbf{A}\mathbf{x} \leq \mathbf{b}\}$ is called the *feasible set*, and represents the set of alternatives or *search space*. Each element $\mathbf{x} \in \mathcal{X}$ is a feasible solution of Equation 3.1. A solution $\mathbf{x}^* \in \mathcal{X}$ is called *optimal*, if $\forall \mathbf{x} \in \mathcal{X}\ z(\mathbf{x}^*) \leq z(\mathbf{x})$ holds. The space, of which $\mathcal{X}$ is a subset, is called *decision space*. The image of $\mathcal{X}$ under $z(\cdot)$ is denoted $\mathcal{Y} := \{z(\mathbf{x}) | \mathbf{x} \in \mathcal{X}\}$ and represents the feasible set in *criterion space*[39]. Integer linear programs can be generalized by relaxing the integrality constraint for a subset of variables $\mathbf{y} \in \mathbb{R}_+^m$. This class is then called mixed integer programs (MILP), which is formulated as follows:

$$\begin{aligned} &\min \mathbf{c}^T\mathbf{x} + \mathbf{h}^T\mathbf{y} \\ &\text{s.t. } \mathbf{A}\mathbf{x} + \mathbf{G}\mathbf{y} \geq \mathbf{b} \\ &\mathbf{x} \in \mathbb{Z}_+^n, \mathbf{y} \in \mathbb{R}_+^p, \end{aligned} \tag{3.2}$$

In the following, a selection of algorithms for solving MILPs is introduced. Most exact methods utilize the efficiently solvable *linear program* (LP) relaxation of a MILP, which disregards the restriction of $\mathbf{x}$ to an integer domain. The solution to the LP relaxation is the *dual bound* of the MILP. The most prominent algorithm to solve LPs is the *simplex algorithm* proposed by Georg Dantzig in 1947[40]. Therefore, we will first introduce the simplex algorithm followed by exact MILP solving algorithms such as the *branch-and-bound*, the *cutting plane*, and the *branch-and-cut* methods.

3.1.1 The Simplex Algorithm

The simplex algorithm was one of the first methods proposed to solve LPs and is the basis of many exact algorithms for MILPs due to its favorable connections to dual theory. While its worst-case runtime is exponential[41], its average-case complexity is polynomial under many probability distributions[42], explaining its observed efficiency in practice. Geometrically, the simplex algorithm moves along the extreme points of the polytope spanned by the half-spaces of the constraints until it reaches a local minimum, which is due to the convexity of LPs also the global minimum (Figure 3.1). This observation can also be obtained

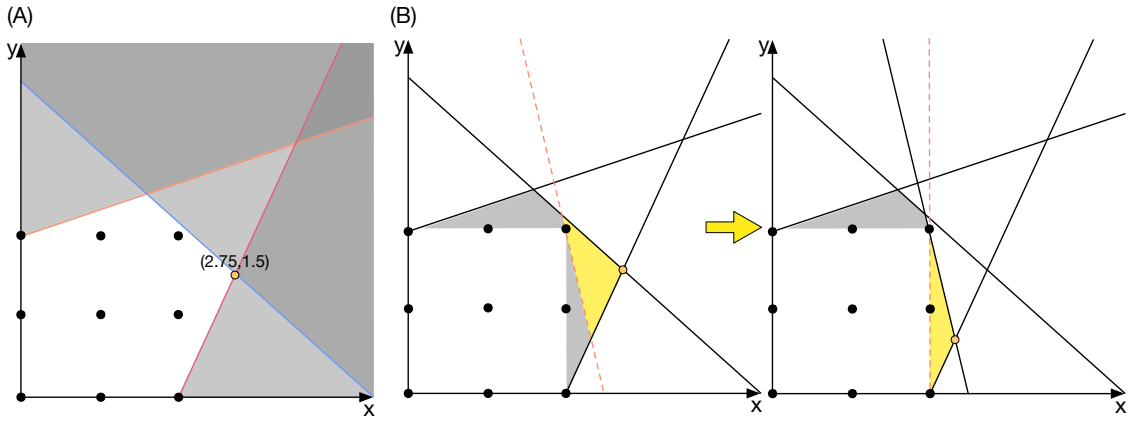


Figure 3.1: (A) Graphical representation of a linear program (LP). Each constraint defines a half-space that restricts the search space of the LP. The dots represent integer solutions of the LP. The simplex algorithm moves along the vertices of the polytope defined by the constraints until the maximum/minimum is reached. (B) Depiction of the cutting-plane method. In each iteration, a new constraint is generated which separates the current LP solution from all integer solutions. The cutting procedure is repeated until an integer solution is obtained by the simplex algorithm.

algebraically. We first note that the constraint matrix $\mathbf{A}$ is w.l.o.g. full row ranked, and can be decomposed into a *basis* $\mathbf{B}$ of l linearly independent columns and in the non-basic columns $\mathbf{N}$ of $\mathbf{A}$, accordingly a feasible solution $\mathbf{x}^*$ is called *basic feasible*, if all variables $x_{\mathcal{B}}$ contained in the basis fulfill $x_{\mathcal{B}} = \mathbf{B}^{-1}b \leq 0$ and all variables that are not contained in the basis $x_{\mathcal{N}}$ are zero. We can show that for each LP that is closed and bounded, there exists at least one basic feasible solution that solves the LP, and that each basic feasible solution corresponds to a vertex of the polytope (and *vice versa*)[43].

Algorithmically, the simplex algorithm is comparable to the Gaussian elimination procedure[44]. In each iteration, the LP is revised into an equivalent form that has some additional structure, which is called *pivoting*. After several iterations and rewriting, the solution can be easily obtained from the transformed LP.

To apply the simplex algorithm, a LP has to be formulated in *standard-slack form* first by introducing *slack variables* $\mathbf{s}$ to transform the inequality constraints into equality constraints:

$$\begin{aligned} &\min \mathbf{c}^T\mathbf{x} \\ &\text{s.t. } \mathbf{A}\mathbf{x} + \mathbf{s} = \mathbf{b} \\ &\mathbf{x} \in \mathbb{R}^n_+, \mathbf{s} \in \mathbb{R}^m_+. \end{aligned} \tag{3.3}$$

In this form the LP is expressed in terms of its basic $\mathbf{x}_\mathcal{B}$ and non-basic variables $\mathbf{x}_\mathcal{N}$ (the non-negative constraint is implicitly assumed in the following):

$$\begin{aligned} &\min \mathbf{c}_\mathcal{B}^T \mathbf{x}_\mathcal{B} + \mathbf{c}_\mathcal{N}^T \mathbf{x}_\mathcal{N} \\ &\text{s.t. } \mathbf{B}\mathbf{x}_\mathcal{B} + \mathbf{N}\mathbf{x}_\mathcal{N} = \mathbf{b} \end{aligned} \tag{3.4}$$

To determine whether the optimum is reached, the objective function is reformulated in terms of its non-basic variables:

$$\begin{aligned} &\mathbf{c}^T\mathbf{x} = \mathbf{c}_\mathcal{B}^T\mathbf{x}_\mathcal{B} + \mathbf{c}_\mathcal{N}^T\mathbf{x}_\mathcal{N} \\ &\text{with } \mathbf{x}_\mathcal{B} = \mathbf{B}^{-1}b - \mathbf{B}^{-1}\mathbf{N}\mathbf{x}_\mathcal{N} \text{ we obtain:} \\ &\quad \Rightarrow \mathbf{c}_\mathcal{B}^T(\mathbf{B}^{-1}b - \mathbf{B}^{-1}\mathbf{N}\mathbf{x}_\mathcal{N}) + \mathbf{c}_\mathcal{N}^T\mathbf{x}_\mathcal{N} \\ &\quad \Rightarrow \mathbf{c}_\mathcal{B}^T\mathbf{B}^{-1}b + \overline{\mathbf{c}}_\mathcal{N}^T\mathbf{x}_\mathcal{N}, \end{aligned} \tag{3.5}$$

where $\overline{\mathbf{c}}_\mathcal{N}^T = \mathbf{c}_\mathcal{N}^T - \mathbf{c}_\mathcal{B}^T\mathbf{B}^{-1}\mathbf{N}$ represents the possible reduction costs of the non-basic variables. If no non-basic variable has negative reduced costs associated, the optimal solution is reached. Otherwise a pivoting operation is applied by selecting a non-basic variable $x_h, h \in \mathcal{N}$ with negative reduced cost and integrating it into the basis. Thus, a basic variable has also to be identified that will exit the basis. Geometrically, increasing x_h corresponds to moving along an edge of the polytope in the direction $\mathbf{d} \in \mathbb{R}^n$, where $d_h = 1$, $\mathbf{d}_\mathcal{B} = -\mathbf{B}^{-1}\mathbf{A}_h$, and $\mathbf{d}_{\mathcal{N}_h} = 0$. For each basic variable $x_{\mathcal{B}_i}$ the maximal step length $\Delta x_{\mathcal{B}_i} = \frac{-x^*_{\mathcal{B}_i}}{d_{\mathcal{B}_i}}$ until $x_{\mathcal{B}_i}$ reaches zero is calculated. The basic variable $x_{\mathcal{B}_k}$ with the minimum step length:

$$x_{\mathcal{B}_k} = \arg\min\{\Delta x_{\mathcal{B}_i} | x_i \in B\} \tag{3.6}$$

is selected as exiting basic variable and set to zero. The selected non-basic variable x_h, is set to $x_h = \Delta x_{\mathcal{B}_k}$ and enters the new basis. All other basic variables are moved into the direction $\mathbf{d}$ with the determined step length $\Delta x_{\mathcal{B}_k}$:

$$\hat{x}_{\mathcal{B}_i} = x_{\mathcal{B}_i} + \Delta x_{\mathcal{B}_k} d_{\mathcal{B}_i}, \ \forall\ i \neq k. \tag{3.7}$$

Duality in Linear Programs

Duality is a very important concept in mathematical optimization, and especially in linear programming. It provides bounds on the original problem and another way to prove optimality. Each LP in standard form, which is also called the *primal problem*, has a complementary *dual problem*. Both problems can be symmetrically transformed into each

other:

$$\begin{array}{lll} \text{Primal:} & & \text{Dual:} \\ \min \quad \mathbf{c}^T\mathbf{x} & & \max \quad \mathbf{b}^T\mathbf{r} \\ \text{s.t.} \quad \mathbf{A}^T\mathbf{x} \geq \mathbf{b} & & \text{s.t.} \quad \mathbf{A}^T\mathbf{r} \leq \mathbf{c} \\ \qquad \mathbf{x} \in \mathbb{R}^n_+ & & \qquad \mathbf{r} \in \mathbb{R}^m_+. \end{array} \tag{3.8}$$

Primal and dual form of an LP are connected via two theorems, called the *weak* and *strong duality theorem*. The weak duality theorem states that for a given primal feasible solution $\overline{\mathbf{x}}$ and a dual feasible solution $\overline{\mathbf{r}}$ of the corresponding dual problem $\mathbf{b}^T\overline{\mathbf{r}} \leq \mathbf{c}^T\overline{\mathbf{x}}$ always holds true. In other words, the dual solution is always a lower bound on the primal solution. Moreover, the strong duality theorem states that for an optimal primal solution $\mathbf{x}^*$, the corresponding dual solution $\mathbf{r}^*$ must be dual optimal and $\mathbf{r}^* = \mathbf{c}^T\mathbf{x}^*$ holds. The *complementary slackness theorem* offers an algorithmic approach to verify that a given primal feasible solution $\overline{\mathbf{x}}$ is optimal and to obtain its corresponding dual feasible solution $\overline{\mathbf{x}}$. It states that the following conditions are necessary and sufficient for $\overline{\mathbf{x}}$ and $\overline{\mathbf{r}}$ to be optimal[44]:

$$\begin{aligned} &\sum_{i=1}^{m} a_{ij}\overline{r}_j = c_j \quad \text{or } \overline{x}_i = 0 \quad \forall\, j \in [1,..,n] \\ &\text{and} \\ &\sum_{j=1}^{n} a_{ij}\overline{x}_i = b_i \quad \text{or } \overline{r}_j = 0 \quad \forall\, i \in [1,..,m] \end{aligned} \tag{3.9}$$

The dual simplex algorithm utilizes the complementary slackness to retain a dual feasible solution during the simplex steps until a primal feasible solution is obtained, and thus the optimum is reached. The dual simplex algorithm is in many cases faster than the primal simplex algorithm, especially when constraints are added iteratively, or $\mathbf{b}$ is changed[41]. The manipulation of the primal LP can lead to infeasibility of the current basic feasible solution; therefore the complete modified primal LP has to be solved again. The dual basic optimal solution, however, remains feasible after the addition of new primal constraints or modification of $\mathbf{b}$, but probably loses optimality. Therefore, a few dual pivoting steps, starting from the current dual basis, usually suffice to find the optimal solution of the modified LP. The iterative introduction of constraints and efficient solving of the modified problem is particularly important in branch-and-bound methods to solve ILPs, which we will discuss in the following section 3.1.2.

3.1.2 Branch-and-Bound

Branch-and-Bound (B&B) is an algorithmic paradigm to solve hard combinatorial optimization problems. The algorithm was first proposed by A. H. Land and A. G. Doig in 1960[45] and has remained the main algorithmic approach to solving NP hard combinatorial problems. For simplicity, we only consider strict integer linear programs in the description of the algorithm, but any of the concepts can be applied to mixed integer linear programming as well.

B&B recursively enumerates the solution space $\mathcal{X}$ by partitioning it into disjoint subsets $\mathcal{X}_i, .., \mathcal{X}_k$, which are solved independently. The recursive process is called *branching*. Usually, the resulting subproblems $\mathcal{X}_i, .., \mathcal{X}_k$ are still hard to solve efficiently, however in the case of integer linear programs, a local lower bound LLB($\mathcal{X}_i$) of the optimal solution can be efficiently computed by solving the *LP relaxation* of the corresponding ILP subproblem. In case the optimal solution $\mathbf{x}_i^* \in \mathcal{X}_i$ of the LP relaxed subproblem is integral, the objective value $z(\mathbf{x}_i^*)$ is an upper bound and $\mathbf{x}_i^*$ is a feasible solution of the original ILP. Whenever $z(\mathbf{x}_i^*)$ is smaller as the lowest upper bound found so far, the objective value is set as new *global upper bound* GUB, and $\mathbf{x}_i^*$ is called *incumbent solution*. If the solution $\mathbf{x}_i^* \subset \mathcal{X}_i$ of the LP-relaxed subproblem is not integral, a fractional variable $x_k \in \mathcal{X}_i$ is selected and two constraints $x_k \leq \lfloor x_k^* \rfloor$ and $x_k \geq \lceil x_k^* \rceil$ are added to the newly branching subproblems of $\mathcal{X}_i$ respectively.

Another important concept of B&B, besides branching, is *bounding*. Bounding enables the algorithm to disregard a potentially large proportion of the search space that provably cannot contain the optimal solution of the ILP. Whenever the optimal solution of a relaxed subproblem $\mathcal{X}_i$ is larger as the current GUB(i.e., LLB($\mathcal{X}_i$) $\geq$ GUB), the subproblem and all its subproblems $\mathcal{X}_j \subset \mathcal{X}_i$ can be disregarded as they cannot contain a globally optimal solution of the ILP. The algorithm terminates when all subproblems are bounded by the current GUB, which proves the global optimality of the incumbent solution.

Obviously, B&B does not change the exponential worst-case runtime complexity of combinatorial problems but works quite efficiently in many real world applications up to a certain problem size.

3.1.3 Cutting Planes

The cutting plane method is another important paradigm in combinatorial optimization. It was first described by Fulkerson and Dantzig for a specific combinatorial optimization problem[46] and later generalized to ILPs and MILPs by Gomory in 1958[47]. The general idea of cutting planes is to efficiently and iteratively find new constraints that separate the integer feasible region from the convex hull of the LP relaxation (Figure 3.1). These so called *cuts* are added to the problem and the LP relaxation is re-optimized until an

integral solution is obtained. The cut-generating methods described by Gomory can be applied to any MILP. We therefore will briefly introduce *Gomory's cuts* in the context of strict integer linear programming.

Gomory's cuts can be directly generated from the primal, non-integral solution $\mathbf{x}^*$ of the LP relaxation. To identify a valid cut separating the integral search space from the non-integral search space, a row of the basic feasible solution

$$x_{\mathcal{B}_i} + \sum_{j \in \mathcal{N}} A_{ij} x_j = \bar{b}_i, \tag{3.10}$$

with $\bar{\mathbf{b}} = \mathbf{B}^{-1}\mathbf{b}$, and $\bar{b}_i$ fractional, is selected. The row is then reformulated in terms of its integral and non-integral parts:

$$x_{\mathcal{B}_i} + \sum_{j \in \mathcal{N}} \lfloor A_{ij} \rfloor x_j + \sum_{j \in \mathcal{N}} (A_{ij} - \lfloor A_{ij} \rfloor) x_j = \lfloor \bar{b}_i \rfloor + (\bar{b}_i - \lfloor \bar{b}_i \rfloor). \tag{3.11}$$

Rearranging the equation so that the integral part is on the left-hand side and the fractional is on the right-hand side yields:

$$x_{\mathcal{B}_i} + \sum_{j \in \mathcal{N}} \lfloor A_{ij} \rfloor x_j - \lfloor \bar{b}_i \rfloor = \bar{b}_i - \lfloor \bar{b}_i \rfloor - \sum_{j \in \mathcal{N}} (A_{ij} - \lfloor A_{ij} \rfloor) x_j. \tag{3.12}$$

Notice that the left-hand side is an integer for any integer solution, while the right-hand side is a fraction < 1. Thus, we can formulate the following inequality which will be satisfied by all solutions of the ILP but not of the current LP relaxation:

$$x_{\mathcal{B}_i} + \sum_{j \in \mathcal{N}} \lfloor A_{ij} \rfloor x_j \leq \lfloor \bar{b}_i \rfloor + (\bar{b}_i - \lfloor \bar{b}_i \rfloor). \tag{3.13}$$

As $(\bar{b}_i - \lfloor \bar{b}_i \rfloor)$ is the only fractional part, the following inequality is also satisfied:

$$x_{\mathcal{B}_i} + \sum_{j \in \mathcal{N}} \lfloor A_{ij} \rfloor x_j \leq \lfloor \bar{b}_i \rfloor. \tag{3.14}$$

By substituting $x_{\mathcal{B}_i} = \bar{b}_i - \sum_{j \in \mathcal{N}} A_{ij} x_j$, and adding an additional slack variable s_k the final Gomory cut

$$s_k + \sum_{j \in \mathcal{N}} (A_{ij} - \lfloor A_{ij} \rfloor) x_j = (\bar{b}_i - \lfloor \bar{b}_i \rfloor) \tag{3.15}$$

is added to the LP relaxation. The modified LP is re-optimized and the cut generation routine is repeated until an integral solution is obtained.

3.1.4 Branch-and-Cut

The Branch-and-Cut (B&C) method is a combination of B&B and cutting-plane methods - in particular Gomory's cuts - and was proposed in the mid-1990's by Cornuéjols *et al.*[48]. Cuts can be added at each branch of the B&B search tree to tighten the calculated bounds, which in turn can lead to significant pruning of subproblems. Despite the NP hardness of MILP, B&C works very efficiently in practice and is often much faster than pure B&B. The runtime, however, is influenced by multiple factors such as size and complexity of the MILP model, the employed branching rule and processing of the subproblems, as well as the choice and number of cuts applied at each subproblem.

3.2 Multiobjective Optimization

Many real-world optimization problems involve multiple, possibly conflicting objectives. Multiobjective optimization (MO), or bi-objective optimization in the special case of two objectives, offers a theoretical foundation to make optimal decisions in these circumstances. Often, no single solution exists that simultaneously optimizes all $p \in \mathbb{N}^{\geq 2}$ objective functions. Thus, in MO the set of all optimal trade-off solutions (or a subset thereof) is wanted. In general a multiobjective optimization problem (MOP) can be defined as

$$\begin{aligned} &\min \mathbf{z}(\mathbf{x}) = (z_1(\mathbf{x}), .., z_p(\mathbf{x})) \\ &\text{s.t.} \quad \mathbf{x} \in \mathcal{X}. \end{aligned} \tag{3.16}$$

All definitions introduced in Section 3.1 can be extended to the multiobjective case. However, the notion of optimality of $\mathbf{z} : \mathcal{X} \to \mathbb{R}^p$ has to be redefined as there exists no canonical order in $\mathbb{R}^p$. Therefore, a MOP additionally has to define an ordered space $(\mathbb{R}^p, \preceq)$ and a mapping function $\theta()$ that maps from $\mathbb{R}^p$ to the order space $(\mathbb{R}^p, \preceq)$[39]. Using this definition, a feasible solution $\mathbf{x}^* \in \mathcal{X}$ is said to be an *optimal solution* of an MOP if $\nexists\, \mathbf{x} \in \mathcal{X},\ \mathbf{x} \neq \mathbf{x}^*$ such that

$$\theta(\mathbf{z}(\mathbf{x})) \preceq \theta(\mathbf{z}(\mathbf{x}^*)). \tag{3.17}$$

Based on the tuple of ordered space and mapping function $\cdot/\theta/(\mathbb{R}^p, \preceq)$, MOPs can be divided into multiobjective optimization classes[39]. In this work, only two multiobjective classes are considered, the class of lexicographical ordered MOPs $(\mathcal{X}, \mathbf{z}(), \mathbb{R}^p)/\operatorname{id}/(\mathbb{R}^p, \leq_{\text{lex}})$ (Chapter 5), and the class of component-wise MOPs $(\mathcal{X}, \mathbf{z}(), \mathbb{R}^p)/\operatorname{id}/(\mathbb{R}^p, \leq)$ (Chapter 6). We will focus on the latter and will only discuss properties of the component-wise MOP in the reminder of this section.

3.2.1 Efficiency and Nondominance

An optimal solution of a component-wise MOP $\mathbf{x}' \in \mathcal{X}$ is called *efficient* or *Pareto optimal*, if $\nexists\ \mathbf{x} \in \mathcal{X}$ such that $z_i(\mathbf{x}) \leq z_i(\mathbf{x}')$ for $i \in [1, .., p]$ and $\mathbf{z}(\mathbf{x}) \neq \mathbf{z}(\mathbf{x}')$. Its corresponding objective vector $\mathbf{z}(\mathbf{x}')$ is called a *non-dominated point*[39]. The set of all efficient solutions $\mathbf{x}' \in \mathcal{X}$ is denoted by $\mathcal{X}_E$, whereas the set of all non-dominated points is denoted by $\mathcal{Y}_N$ and often referred to as *Pareto front* (Figure 3.2 (A)). Finding the whole Pareto front of a given MOP is the goal of multiobjective optimization. In the context of MOLPs *supported efficient solutions* become relevant. An efficient solution $\mathbf{x}' \in \mathcal{X}_E$ is called supported efficient if there exists a $\lambda \in \mathbb{R}^p_+$ such that $\mathbf{x}'$ is an optimal solution to the following transformed single objective optimization problem:

$$\min_{\mathbf{x} \in \mathcal{X}} \lambda^T \mathbf{z}(\mathbf{x}). \tag{3.18}$$

The corresponding objective vector $\mathbf{z}(\mathbf{x}')$ is called *supported non-dominated point* and is located on the convex hull of the Pareto front[49]. Efficient solutions of an MOLP are all supported and also connected, i.e. all points on a line between two consecutive non-dominated points are also non-dominated[50]. Thus the whole Pareto front of an MOLP can be sufficiently described by the extreme points of its convex hull.

3.2.2 Scalarization Methods

The classical approach to solving MOPs transforms the MOP into a parameterized single objective problem that is solved multiple times with altering parameters. Such methods are summarized under the term *scalarization*.

The most prominent and widely used scalarization technique due to the MOLP properties discussed in Section 3.2.1 is the so-called *weighted-sum method* (Figure 3.2 (B)). The weighted-sum method combines the different objectives into a weighted sum

$$\min_{\mathbf{x} \in \mathcal{X}} \sum_{i=1}^{p} \lambda_i z_i(\mathbf{x}), \quad \text{with } \lambda \in \mathbb{R}^p_{>0} \tag{3.19}$$

and finds all *extreme supported non-dominated points* by altering these weights.

A second regularly applied scalarization method, usually in the context of MOILPs, is the ϵ-constraint method[51]. It focuses on one objective and adds all other objective functions as constraints with an upper limit to the problem formulation (Figure 3.2 (C)).

$$\begin{aligned} &\min_{\mathbf{x} \in \mathcal{X}} z_j(\mathbf{x}) \\ &\text{s.t. } z_i(\mathbf{x}) \leq \epsilon_i \quad \forall\ i \in [1, ..p], i \neq j \end{aligned} \tag{3.20}$$

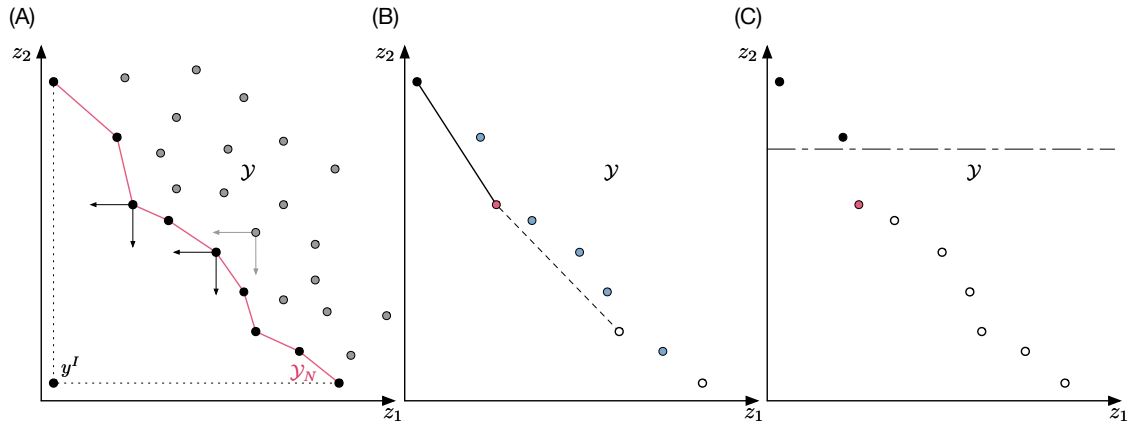


Figure 3.2: (A) Biobjective minimzation example. The points on the red line constitute the Pareto front. All points behind the front are feasible solutions but are dominated by at least one point on the Pareto front (indicated by the vertical and horizontal lines). y^I represents the ideal point where both objectives would become minimal if it were obtainable. (B) The weighted sum approach linearly combines the objective functions to a single-objective optimization problem, but therefore is only able to find non-dominated points residing on the convex hull of the Pareto front (dotted line and white points). (C) The ϵ-constraint method optimizes only one objective and adds all other with an upper bound as constraints to the single-objective model. By iteratively changing the upper bounds - indicated by the dotted line - all non-dominated points can be obtained.

It iteratively solves the single-objective model and subtracts a small, constant value ϵ_i from the constrained objectives to obtain a new non-dominated point in the next iteration.

Other methods combine both approaches[52] or relax the objective constraints[53]. The last method presented here is the *augmented weighted Tchebycheff method* that is applicable to linear MOPs[54–56]. It searches for non-dominated points by minimizing the distance to the ideal point $\mathbf{y}^I = \mathbf{z}^I(\mathbf{x})$ defined by $\mathbf{y}_k^I = \min_{\mathbf{x}\in\mathcal{X}} \mathbf{c}_k^T\mathbf{x}$:

$$\min_{x\in\mathcal{X}} \max_{k=1}^{p} \nu_k(\mathbf{c}_k^T\mathbf{x} - \mathbf{y}_k^I) + \gamma \sum_{k=1}^{p} (\mathbf{c}_k^T\mathbf{x} - \mathbf{y}_k^I), \tag{3.21}$$

where $\nu > 0$ and $\gamma > 0$ are usually chosen as small positive weights. By altering the parameters ν and γ, all non-dominated points can be obtained.

3.2.3 Multiobjective Integer Programming

Multiobjective integer (MOILP) and mixed integer linear problems (MOMILP) are in many ways much harder to solve than MOLPs. This is not only attributable to the NP-hardness of general MILPs, but also to the existence of so-called *unsupported non-dominated points*, i.e. non-dominated points that do not lie on the convex hull of the Pareto front.

Due to the existence of these unsupported non-dominated points, simple scalarization techniques such as the weighted-sum approach do not yield the complete non-dominated set. The ϵ-constraint method and derivatives thereof, on the other hand, are able to recover the unsupported non-dominated points as they usually do not rely on convexity assumptions. But the constraints on objective values usually render the problem NP-hard, even if the objective function can be solved in polynomial time[39]. The augmented weighted Tchebycheff method is also able to retain all non-dominated points, but has to be reformulated in order to linearize the max-term by introducing new constraints based on parts of the objective function. This again suggests that the augmented weighted Tchebycheff method is also NP-hard[39].

Chapter 4

NGS-based HLA Genotyping using Combinatorial Optimization

Parts of this chapter were published in:

Szolek, A.,* ***Schubert, B.****, Mohr, C.*,* et al. *(2014).*
OptiType: precision HLA typing from next-generation sequencing data.
Bioinformatics, 30(23), 3310-3316.

* Joint First Authors.

4.1 Introduction

The human leukocyte antigen (HLA) cluster is one of the most important loci of the adaptive immune system. It encodes for all major and minor HLA class I and II molecules, which present small, potentially immunogenic peptides to $CD4^+$ or $CD8^+$ T-cells to induce an immune reaction. Hence, the identification of a person's HLA genotype is of interest in many biomedical applications, such as vaccinology[57,58], regenerative and translational medicine[59,60], and in autoimmune disease-related research[61,62]. Due to its high polymorphic variation and strong sequence similarity between alleles and even across loci, standard short-read sequencing based methods are ill-equipped to identify the HLA genotype of an individual unambiguously. Alignment-based genotyping methods cannot be used due to massively ambiguous read alignment (Figure 4.1), as well as the absence of a suitable reference sequence. Therefore, established methods make use of labor- and time-intensive techniques constructing allele-specific oligonucleotide probes for hybridization, or they use allele-specific primers for PCR amplification[63]. Other, non-sequence based methods use specifically designed antibodies to identify HLA isoforms[63].

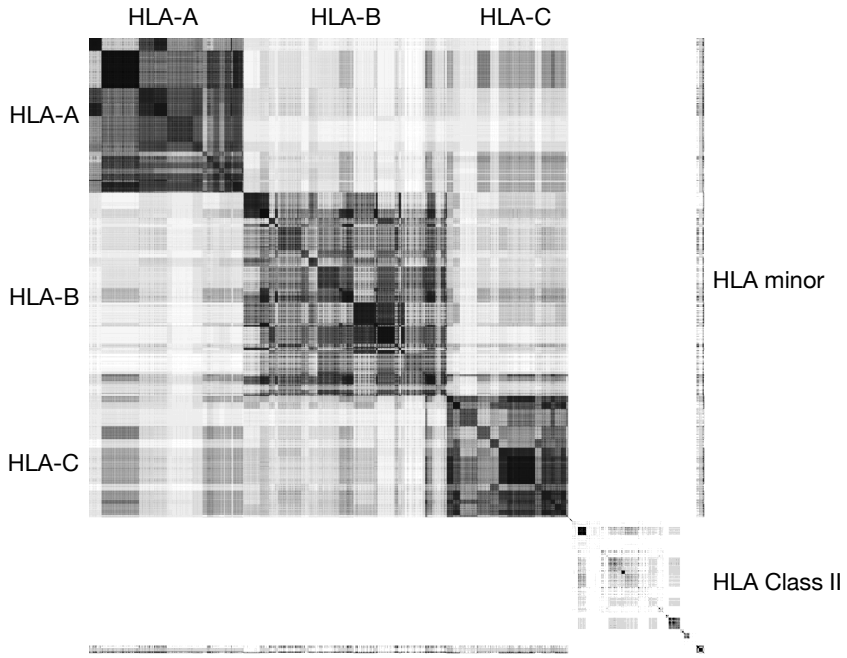


Figure 4.1: HLA read alignment matrix. Reads for each known HLA allele were constructed and aligned against all other HLA alleles. Each row and column represents an HLA allele and the entry represents how many reads of one HLA allele mapped to another.

In recent years, new NGS-based sequencing protocols were proposed, overcoming some of the very time-consuming steps of older methods[64–67]. These methods remain labor-intensive with a turnover of approximately two days[67] and artificially amplify the HLA locus. However, using algorithmic solutions addressing the deconvolution of read ambiguities and the reference problem on standard non-amplified NGS data, it should be theoretically possible to determine the HLA genotype of an individual without the need to generate additional data for the sole purpose of HLA genotyping. Such an approach would decrease time and cost expenses, since in many clinical centers the sequencing of patients has become standard practice for diagnostic purposes.

Related Work

The first computational work on the topic was proposed by Erlich *et al.* in 2011[68]. It was based on posterior probability estimation of allele pairs and was fully integrated into a 454 Titanium sequencing pipeline[68]. Two years later, Warren *et al.* published the first algorithmic solution, called *HLAminer*, that was able to infer the HLA genotype without the need to specially tailor NGS-pipelines[69]. *HLAminer* used *de novo* assembly to overcome the problem of the non-existing alignment reference and used an allele-specific scoring based on the aligned contigs. The HLA alleles with the highest scoring of each locus assembled

the HLA genotype of the individual. In 2012, Bögel *et al.* proposed an alignment-based greedy algorithm, called *seq2HLA*. The authors used all known HLA sequences and aligned reads against the so-constructed set of reference sequences. The first HLA alleles are independently determined for each locus based on the number of reads aligned to it. In a second step, a potential second heterozygous allele is selected that explains a significant amount of unassigned reads[70]. Both methods, *HLAminer* and *seq2HLA*, exemplify the two strategies utilized to overcome the high polymorphism of the HLA cluster. *ATHLATES*, proposed by Liu *et al.*[71], combined the reference approach with *de novo* assembly to select HLA allele pairs with the smallest Hamming distances of the aligned contigs to known exonic regions of all HLA alleles. Others used complicated tree structures to deconvolute the aligned reads (*HLAforest*[72]), or used filtering criteria based on coverage depth and base coverage as pre-processing step[73]. All methods showed moderate accuracy in benchmark studies. *seq2HLA* was only able to accurately predict two-digit HLA genotypes; *HLAminer* and *HLAforest* achieved an accuracy of 85-90% correctly predicted four-digit HLA genotypes on RNA-Seq data. For short-read RNA-Seq and WGS data, the performance was even lower. Major *et al.* were able to yield a performance of 94% correctly predicted four-digit HLA genotypes on exome data that fulfilled all their strict quality criteria, but had to omit a significant percentage of samples of their test set as they did not fulfill their quality criteria. *ATHLATES* claimed to produce predictions with 99% accuracy, but was only tested on 15 samples of which only 11 were publicly available. Thus, the performance cannot be assessed independently and might be overestimated.

A major factor of the poor performance might be the independence assumption made to deconvolute the ambiguously aligned reads. All mentioned methods either treat each allele or each locus independently and, therefore, count ambiguous reads mapping across alleles and loci multiple times. Also, an unexploited source of information for WGS and exome date are intronic sequences. A study by Blasczyk *et al.* showed that the observed polymorphic variability in intronic segments stems from highly systematic mutations reflecting the ancestral lineage of the allele[74]. Thus, harnessing intronic information could increase prediction performance. However, incorporating intronic sequences is non-trivial, as 94% of all known HLA alleles contained in IMGT[32] - the largest database of HLA sequences - are incomplete.

Project Overview

To overcome the depicted problems and include the available intronic information, we developed a new alignment-based algorithm called *OptiType*, that simultaneously selects all major and minor HLA class I alleles to maximize the number of reads that can be explained by the predicted genotype, thereby adequately addressing the problem of cross-allele, and

cross-loci mapped reads. A read is said to be explained by an inferred genotype if the

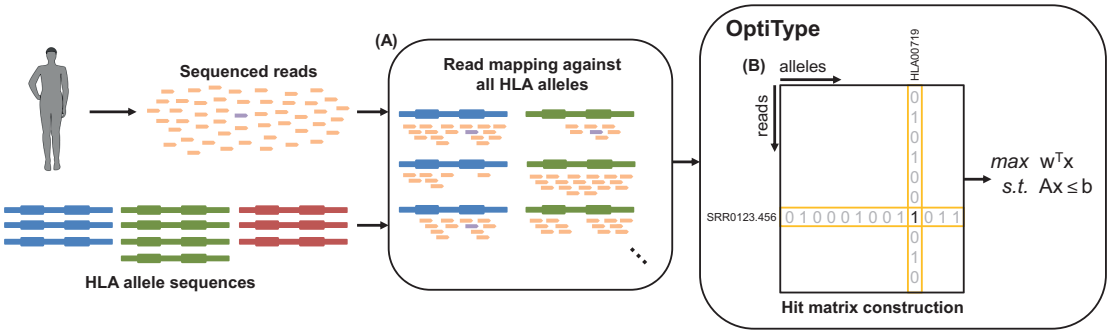


Figure 4.2: OptiType's four-digit HLA typing pipeline. Reference libraries for genomic and CDS sequences are generated by extracting exons 2 and 3 from each known HLA-I allele. For genomic sequences, flanking intronic regions are also extracted. If some of these regions are missing, phylogenetic information is used to reconstruct the missing segments from the closest relative HLA-I allele. NGS reads are mapped against the so-constructed HLA allele reference (A). From the mapping result, a binary hit matrix $\mathbf{C}^{R\times H}$ is constructed for all reads $r \in R$ mapping to at least one allele $h \in H$ of the reference with $\mathbf{C}_{r,h} = 1$ iff read r could be mapped to allele h otherwise, $\mathbf{C}_{r,h} = 0$ (B). Based on this hit matrix, an ILP is formulated that optimizes the number of explainable reads by selecting up to two alleles (columns of the hit matrix) for each HLA-I locus (C). The selected alleles represent the most probable genotype

read aligns to at least one HLA allele of the genotype with no more mismatches than to any other allele. This formulation circumvents the problem of considering ambiguously mapped reads multiple times during inference but comes with the cost of a combinatorial explosion of potential HLA allele combinations. To efficiently solve the stated HLA genotype inference problem, we embedded it into a well-studied combinatorial problem, namely into a particular case of the set covering problem. The optimization form of the set covering problem selects up to k sets S_i to maximize the size of their union $\cup_{i\in I} S_i$ with S_i being a subset of a universe U. Here, the universe U consists of all mapped reads, and a subset is characterized by an HLA allele and the reads explained by it. For each locus either one (homozygous case) or two (hehterozygous case) HLA alleles can be selected simultaneously. The algorithm follows three steps to construct the set covering instance for HLA genotype inference (Figure 4.3). First, reads are mapped against an HLA allele reference set. Since only exon 2 and 3 have been fully sequenced for all HLA-I alleles, we considered only these regions during mapping so that no allele is disadvantaged due to missing sequence information (Figure 4.3 A). For exome and WGS data, we additionally included intronic sequences flanking exon two and three and developed a phylogenetic-based schema to impute missing intronic segments. From these mapped reads a binary hit matrix was constructed in a second step (Figure 4.3 B). Each entry of the matrix indicated whether a

particular read could be explained by an HLA allele or not. Based on this hit matrix, a set covering instance is formulated and solved as integer linear program (Figure 4.3 C).

4.2 Materials and Methods

4.2.1 Reference Construction

HLA exon coding sequences (CDS), genomic sequences, as well as annotations were extracted from the IMGT/HLA database (Release 3.14.0, July 2013[32]). The RNA reference database was constructed by concatenating exon 2 and 3 of all HLA alleles. For exome and WGS data, flanking intronic sequences were also considered. Partially sequenced HLA alleles with missing intronic sequences were reconstructed by imputing the closest, fully sequenced alleles with respect to sequence similarity based on k-tuple measure[75] calculated by ClustalOmega[76] 1.1.0. On average, partial alleles had 1.6($\pm$1.04) fully sequenced nearest neighbors with unique intron sequences. For sequences with multiple nearest neighbors, all possible combinations were generated resulting in 10,779 reconstructed sequences for 6,489 partial alleles[77]). A leave-one-out cross validation was performed using the fully sequenced HLA alleles to validate the quality of the reconstruction procedure. Exon 1, 2, and 3 of one allele were discarded and reconstructed based on the remaining once. The sequence similarity of the reconstructed and original HLA allele was 99.89%($\pm$0.43%), corresponding to an average 1.2 edit distance error on the three introns combined. For comparison, sequence similarity between introns of the same loci was found to be 97.36% ($\pm$2.15%), corresponding to 29 bp differences on average.

4.2.2 Read Mapping

Read mapping was performed with *RazerS3*[78] 3.1. All best alignments for every read with a sequence identity of at least 97% were taken into account (*–percentidentity 97 –distance-range 0*). The maximum number of reported best matches (*–max-hits*) was set to infinity. All read matches fulfilling those criteria were reported and stored.

4.2.3 Hit Matrix Construction

Based on the aligned reads $r \in R$ to HLA alleles $h \in L$, a binary hit matrix $\mathbf{C}^{R \times L}$ was constructed with $\mathbf{C}_{r,h} = 1$ iff read r could be mapped to HLA allele h; otherwise $\mathbf{C}_{r,h} = 0$. The columns of rare alleles that were not reported in *allelefrequency.net*[79] or dbMHC[80] were removed. Identical rows, resembling reads with the same mapping profile, were collapsed and represented by a weighting vector $\mathbf{o}_r$. To further reduce the dimensionality, alleles were identified that were unlikely part of the HLA genotype. An allele was deemed unlikely, if all its explained reads could have originated from another allele which explained additional

reads. As a result, these identified alleles were deleted, since their biological evidence could be explained by another, more likely HLA allele. More formally, columns for which $(\mathbf{C}_{:,h}^T \mathbf{C}_{:,g} = \mathbf{C}_{:,h}) \wedge (|\mathbf{C}_{:,h}| < |\mathbf{C}_{:,g})$ with $h, g \in L$ hold true, were dropped as these columns never could be part of an optimal solution.

4.2.4 Formulation of the Set Covering Problem

We base our formulation of the HLA genotype inference problem on the premise that the correct genotype also explains the majority of mappable reads. Thus, we are searching for a combination of up to six major and minor HLA-I alleles that maximize the number of mappable reads under the biological constraints that at least one and at most two alleles are selected per locus (constraint C1 and C2 in Eq. 4). Constraint C1 and C2 reflect the diploid nature of the human genome and allow for homozygosity in the genotype. Such problems can be conveniently modeled as a set covering problem, which in turn can be expressed and solved as an integer linear program (ILP). In the following, we derive the ILP representing the set covering formulation for HLA-I genotype inference.

We introduce a binary variable x_h for each HLA allele $h \in L$ with $x_h = 1$ iff allele h is part of the optimal HLA genotype $S \subseteq L$. Another binary variable y_r, representing each read of matrix $C^{R \times L}$, is additionally introduced. Based on the hit matrix, a constraint (C3) is formulated that forces $y_r = 1$ iff read r could originate from the current genotype S (i.e., it could be mapped to at least one of the alleles $h \in S$). With these formulations we arrive at the following ILP:

$$\text{(O1)} \quad \max_{\mathbf{y}} \sum_{r \in R} o_r \cdot y_r \tag{4.1}$$

s.t.

$$\text{(C1)} \quad \forall\, X \in \{A, B, C, G, H, J\} \qquad \sum_{h \in X} x_h \leq 2$$

$$\text{(C2)} \quad \forall\, X \in \{A, B, C, G, H, J\} \qquad \sum_{h \in X} x_h \geq 1$$

$$\text{(C3)} \quad \forall\, r \in R \qquad \sum_{h \in L} C_{r,h} \cdot x_h \geq y_r$$

with o_r being the number of previously collapsed rows with the same mapping profile, and A, B, C, G, H and J the sets of alleles for the major loci HLA-A, B, C and the minor loci HLA-G, H, J.

While this formulation favors heterozygous loci combinations due to spurious hits caused by sequencing errors and other sources of error, it is necessary to correct the objective

function with a regularization term $g(r)$ to account for homozygosity:

$$g(r) = \begin{cases} \sum_{h \in L} x_h - n^{loci} & \text{if } y_r = 1 \\ 0 & \text{otherwise} \end{cases}$$

where n^{loci} describes the number of loci. The regularization term is weighted by a constant β representing the proportion of reads that have to be additionally explained by an allele combination to choose a heterozygous solution over a homozygous one. The defined regularization term can be integrated into the existing ILP by introducing an additional integer variable g_r for each read $r \in R$ and three constraints (C4-C6). In addition, we introduce a small penalizing constant γ to prioritize alleles with full sequence information over reconstructed alleles contributing to equally good solutions. The final ILP formulation is thus given by:

$$\text{(O1)} \quad \max_{\mathbf{y}} \sum_{r \in R} o_r(y_r - \beta g_r) - \gamma \sum_{h \in L^R} x_h \tag{4.2}$$

s.t.

$$\text{(C1)} \quad \forall\, X \in \{A, B, C, G, H, J\} \qquad \sum_{h \in X} x_h \leq 2$$

$$\text{(C2)} \quad \forall\, X \in \{A, B, C, G, H, J\} \qquad \sum_{h \in X} x_h \geq 1$$

$$\text{(C3)} \quad \forall\, r \in R \qquad \sum_{h \in L} C_{r,h} x_h \geq y_r$$

$$\text{(C4)} \quad \forall\, r \in R \qquad g_r \leq \tau^{\text{loci}} y_r$$

$$\text{(C5)} \quad \forall\, r \in R \qquad g_r \leq \sum_{h \in L} x_h - n^{\text{loci}}$$

$$\text{(C6)} \quad \forall\, r \in R \qquad g_r \geq (\sum_{h \in L} x_h - n^{\text{loci}}) - n^{\text{loci}}(1 - y_r)$$

where $L^R \subseteq L$ is the set of reconstructed alleles, and γ is a small constant factor penalizing the use of reconstructed alleles ($\gamma = 0.01$).

β was fit with five-fold cross-validation on 230 samples of the 1,000 Genomes project. The cross-validation folds have been stratified regarding homozygous and heterozygous genotypes. Best performance was achieved with $\beta = 0.09$.

4.2.5 NGS Test Data Sets

To allow comparison with other published methods, the same publically available samples have been used to evaluate *OptiType*. We therefore extracted 16 colorectal cancer RNA-Seq

[SRP010181], and 20 low-coverage WGS samples of the HapMap Project[81] from the NCBI Sequence Read Archive[80]. Both data sets have been used by Warren *et al.* and Kim *et al.* and contained pair-end samples with 100 to 102 bp long reads[69,72]. Additional 50 lymphoblastic cell line short read RNA-Seq samples [ERA002336][82], as well as 12 WGS samples of the HapMap Project have been extracted to be comparable with Boegel *et al.* and Major *et al.*[70,73]. The short-read RNA-seq samples were pair-end and contained 37 bp long reads. Furthermore, 161 exome sequencing samples of the 1,000 Genomes Project[83] were obtained, which have been used by Major *et al.*[73], and extended to 253 exome sequencing samples generated by Illumina HiSeq 2000 and Genome Analyzer II of the 1,000 Genomes Project. The eleven used samples to evaluate ATHLATES[71] have been part of the extended 1,000 Genomes data set. A list of the used samples with accession IDs is given in Appendix Table F.1.

4.2.6 Performance Metric

We used the percentage of correctly predicted HLA alleles and loci per sample as performance measure, as it was already used by Boegel *et al.* and Warren *et al.*. The percentage of correctly predicted zygosity was used as a second independent performance measure. We defined the zygosity of a locus to be accurately predicted if it matched the experimental zygosity without considering whether the HLA alleles were correctly predicted.

4.2.7 Implementation

OptiType was implemented in Python 2.7 using Pandas with HDF 5 support. The necessary read mapping was performed with RaserzS3[84] and Bowtie2[85]. The ILP was formulated in Pyomo 3.3[86], a Python-based modeling language, and solved with CPLEX 12.5 (`www.ilog.com`). OptiType is published under a three-clause BSD license and available at `http://github.com/FRED-2/OptiType`.

The statistical analysis was performed in R 3.0.2. Bootstrapping with 100,000 repetitions was used to calculate 95% confidence intervals. Statistical comparisons have been carried out with a one-sided sign test if not stated otherwise and considered significant at a significance level of 0.05.

4.3 Results

To establish *OptiType's* overall performance, we tested its performance on a large heterogeneous data set consisting of WGS, WES, and RNA-Seq data of varying read lengths. We then compared *OptiType's* performance with previously published methods on subsets used in the respective publications and established the benefits of our intron-reconstruction

approach. Our analysis was concluded with a sensitivity study on the performance influence of read length and coverage depth, two main factors we believed to be influential on prediction accuracy.

4.3.1 Overall Accuracy and Comparison

OptiType was tested in total on 361 samples and achieved an accuracy of 97.1% (CI_{95} 96.1 − 97.8%) on four-digit level, and 99.3% (CI_{95} 98.7 − 99.7%) on two-digit level, correctly predicting 939 of 950 (98.8%) heterozygous and 127 of 133 (95.5%) homozygous loci.

OptiType significantly outperformed *HLAminer*, *seq2HLA*, *HLAForest*, and the approach of Major *et al.* on the data sets the different methods were evaluated on respectively (Figure 4.3). The accuracy gain amounted to 4 to 15%, corresponding to a 65 to 83% decrease of incorrectly predicted alleles. Only *ATHLATES* showed comparable performance on its small benchmark set consisting of 11 samples.

On the complete 1,000 Genome data set, *OptiType* achieved an average accuracy of 97.6% (CI_{95} 96.7 − 98.4%), with 667 of 676 (98.7%) correctly predicted heterozygous and 80 of 83 (96.4%) correctly predicted homozygous loci.

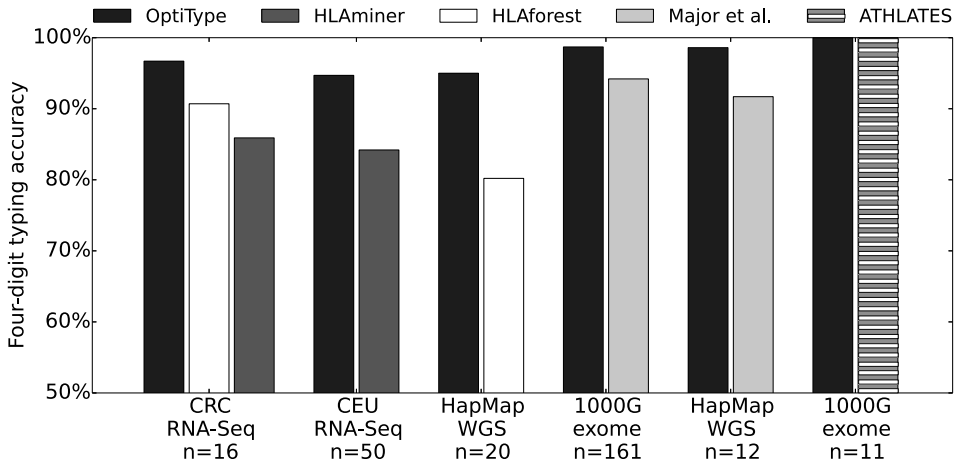


Figure 4.3: Performance comparison of HLA typing algorithms. OptiType's average prediction accuracy for major HLA-I loci was compared with four other published HLA typing methods capable of four-digit typing on publicly available datasets previously used to evaluate these methods

4.3.2 Influence of Intronic Reconstruction

To evaluate the influence of intronic sequence reconstruction for WGS, and WES data, a reference database was constructed using only exon 2 and 3. Instead of using RazerS3 for read mapping, we used Bowtie2 with enabled local alignment (soft clipping) to avoid losing

reads at exonic boundaries. Mismatch tolerance was similar to that of RazerS3's settings (Section 4.2.2).

As the length of exon 2 and 3 is ~ 275 bp, paired-end mapping is complicated. Hence, only one mate could be mapped of a significant amount of paired-end reads. To account for this, we developed two approaches for confusion matrix construction. The first method strictly used pairs of reads, where only matched mates were used, whereas the second also used reads with unmappable mates.

These configurations were tested on the 1,000 Genomes data set and compared to the performance of *OptiType* with sequence reconstruction. On average, *OptiType* yielded an accuracy of 93.5% (CI_{95} $91.8 - 95.1\%$) with the strict matrix construction rule and 90.6% (CI_{95} $89.0 - 92.3\%$) with the hybrid approach of using single-end hits as well. These results correspond to a 2.7- to 3.9-fold increase in error compared to *OptiType* with intronic sequence reconstruction.

4.3.3 Influence of HLA Enrichment and Coverage Depth

As to study the effects of specific HLA enrichment on accuracy, we tested *OptiType* on a WES sample that was prepared with a custom SureSelect HLA enrichment kit provided by Michael Wittig (Institute of Clinical Molecular Biology, Christian-Albrechts-University of Kiel, Germany) and Agilent Technologies. The sample additionally was prepared with a SureSelectXT Human All Exon V5 kit (Agilent Technologies, Böblingen, Germany) and sequenced with an Illumina HiSeq 2500 with 101 bp long reads. Due to the HLA enrichment, the HLA-I loci exhibited an average coverage depth of $\sim 4,100\times$.

From this sample, a subset of reads with decreasing sizes was extracted to simulate different coverage depths. *OptiType* achieved a fully correct genotype prediction with as little as 0.3% of the total number of reads, corresponding to $\sim 12\times$ coverage depth. This amount equals $\sim 15\%$ of reads of a standard, non-HLA enriched WES sample.

To determine the influence of coverage depth on a broader sample, we used all 1,000 Genomes Project exome sequencing samples in a similar manner. Different amounts of reads were randomly re-sampled over 4,000 times to simulate different coverage depths. The reads were additionally trimmed to 2×32 bp to study the influence of read length as well (Figure 4.4). An accuracy of 95% could be achieved with $10\times$ coverage depth while no effect on accuracy could be attributed to the shortened reads.

Overall the runtime of *OptiType* was manly influenced by coverage depth. On average WES samples, *OptiType* has an approximate runtime of 90 minutes[87], of which the majority of time is spent in the read-mapping and hit matrix construction steps. A solution to the set covering problem, however, can usually be obtained in a few minutes if not seconds.

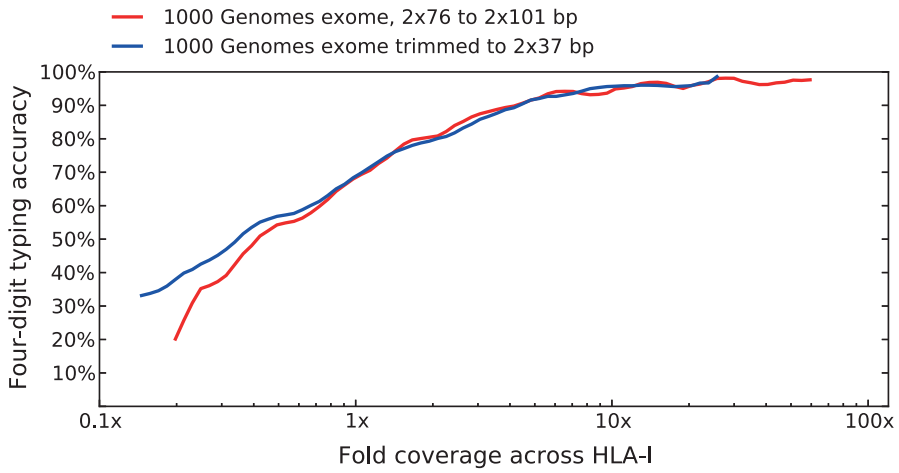


Figure 4.4: Coverage and read length dependence of prediction accuracy. To determine the influence of coverage depth on HLA typing accuracy, reads of 253 exome sequencing runs of the 1,000 Genomes Project were subsampled $> 4,000$ times to simulate different coverage depth conditions. To investigate the impact of read length on performance, original reads were trimmed to 37 bp and evaluated with the same subsampling procedure. Read length alone shows little effect on prediction accuracy and an average coverage depth greater than $10\times$ over the HLA-I loci has been found to already yield maximal accuracy

4.4 Discussion

The HLA genotype is of great importance for many biomedical applications. Standard technology to deduce the HLA genotype of a patient is very time-consuming and costly since it involves the development of custom primers to enrich the HLA region artificially during sequencing. As sequencing of patients is becoming more and more routinely applied in clinics, NGS data not primarily generated for HLA genotyping become a cost and time effective alternative to standard methodologies. Previous attempts also proved that, with the help of algorithmic solutions, it is possible to infer the HLA genotype based on these data[68–72]. The accuracy of these early attempts, however, was unsatisfactory, mostly due to drastic assumptions, such as the independence of all HLA alleles, made in the models to infer the most probable genotype. We therefore developed a new method called *OptiType*, that overcomes these drawbacks and over-simplifications of these early models. *OptiType* is fully automated and infers the HLA type with four-digit resolution on NGS data from RNA-Seq, WES, and WGS technologies. On an extensive benchmark, it demonstrated its superiority to previously published *in silico* HLA typing by significantly outperforming them on both two- and four-digit resolution with an accuracy of 99.3% (CI_{95}: 98.7-99.7%) and of 97.1% (CI_{95}: 96.1-97.80%) respectively. The latter is especially important in clinical applications like individualized vaccine design, prevention of graft-versus-host disease and treatment of

autoimmune diseases. Additionally, *OptiType*, as an *in silico* approach, provides the benefits of great cost reduction and a decrease of turnaround time in comparison to state-of-the-art experimental HLA typing methods. In terms of zygosity prediction, *OptiType* achieved an accuracy of 98.4% (CI_{95}: 97.5-99.1%) on 361 benchmarked runs, correctly predicting 939 of 950 heterozygous loci and 127 of 133 homozygous loci. Since its publication, *OptiType's* performance was again validated in two independent studies by Shukla *et al.*[88] in 2015 and Bauer *et al.*[87] in 2016 and remains the most accurate HLA genotype inference method for HLA-I genotypes available.

In general, coverage depth, as seen in the enrichment and simulation studies, does not play a major role above a certain level. As previously observed by Major *et al.*, the number of covered bases has a stronger influence on the prediction outcome than coverage depth. Short reads, while increasing the complexity of the problem because of higher mapping ambiguities, did not have a negative effect on our method's performance.

Incorrect predictions were mostly found to be caused by three distinct issues. First, sequence stretches not covered by any reads can make it impossible to resolve the ambiguity between the correct allele and alleles differing only on the uncovered segments. Second, zygosity detection occasionally fails in cases where alleles with high sequence similarity constitute a heterozygous locus. In such cases, including both alleles in the solution has little impact on the total number of explained reads compared with including just one of them; therefore, *OptiType* favors the homozygous solution. This problem is normally encountered if the two alleles' distinguishing segments have considerably lower coverage than the rest of their sequence. Third, while typing minor loci generally helps with finding the actual source of reads, mapping to both minor and major loci does not always resolve all ambiguities for every genotype. Additionally, experimental typings of the benchmark datasets were sometimes found to be inaccurate, as also observed for the 1,000 Genomes Project samples[68]. This limits the concordance that can be achieved on these datasets.

It is important to ensure an equal *a priori* chance for every allele to be identified by minimizing the disadvantage of alleles with only partial sequence information. Therefore, only exons 2 and 3 and their flanking intron sequences were used as a reference, reconstructing unknown intron sequences with a phylogeny-based approach for incomplete alleles. Including intron sequences not only helped to retain more read pairs, but information from intronic hits was found to be beneficial to performance increasing *OptiType's* accuracy by 4.5% corresponding to a 2.7-fold decrease in error. Furthermore, with an increasing number of completely sequenced HLA alleles, the used reference sequences could be extended beyond regions surrounding exons 2 and 3, reducing ambiguities and increasing prediction accuracy of *OptiType*. Also the underlying integer linear model of *OptiType* can be easily extended to Type II HLA alleles by incorporating the slightly different biological constraints of HLA-DRB, -DQ, and -DP. As to overcome the *OptiTypes* current bottleneck,

the initial read alignment and hit-matrix construction, pre-build k-mer indexing structures as used in modern RNA-Seq expression analysis algorithms [89,90] could be used, which would in turn decrease the runtime of *OptiType* tremendously, with no to minimal expected loss in accuracy, as could be seen from the read length sensitivity analysis.

To summarize, *OptiType* is a fast and accurate HLA typing method based on NGS data, which provides an alternative approach to common HLA genotyping methods. It can be easily adapted to predict genotypes for loci other than HLA-I such as HLA-II and transporter associated with antigen processing. Nevertheless, the predictions are restricted to the used reference and, therefore, can predict only known alleles.

Chapter 5

Designing String-of-beads Vaccines with Optimal Spacers

Parts of this chapter were published in:

Schubert, B.* *& Kohlbacher, O. (2016).*
Designing String-of-beads Vaccines with Optimal Spacers.
Genome Medicine, ***8****(1), 1.*

5.1 Introduction

The invention of vaccines is one of the greatest achievements in human medical history and has led to the eradication of many infectious diseases[91]. Vaccines work on the premise of inducing long-term memory cells of the adaptive immune system without inducing an actual illness to quickly mount an immune response against future infections of a specific pathogen. Traditional vaccines typically consist of whole attenuated or dead pathogens, a selection of antigens or parts of antigens administered as whole proteins, or DNA/RNA fragments[92–95]. Although these traditional vaccines have been very successful, they bear risks of reversion to infectiousness, especially attenuated, living vaccines. The development and production of traditional vaccines is also complex and costly[96]. As the knowledge of our immune system and the mechanics of infections grew, novel rational-based vaccine design strategies emerged reducing the vaccines to the parts that are necessary to induce the wanted immune reaction. The center of this *rational-based vaccine design* approach are *epitope-based vaccines* (EV). EV use only epitopes, small immunogenic regions of antigens, to induce an immune reaction. The selection of these epitopes is very flexible and can be tailored to fit the pathogen's molecular characteristics and those of the patient, making

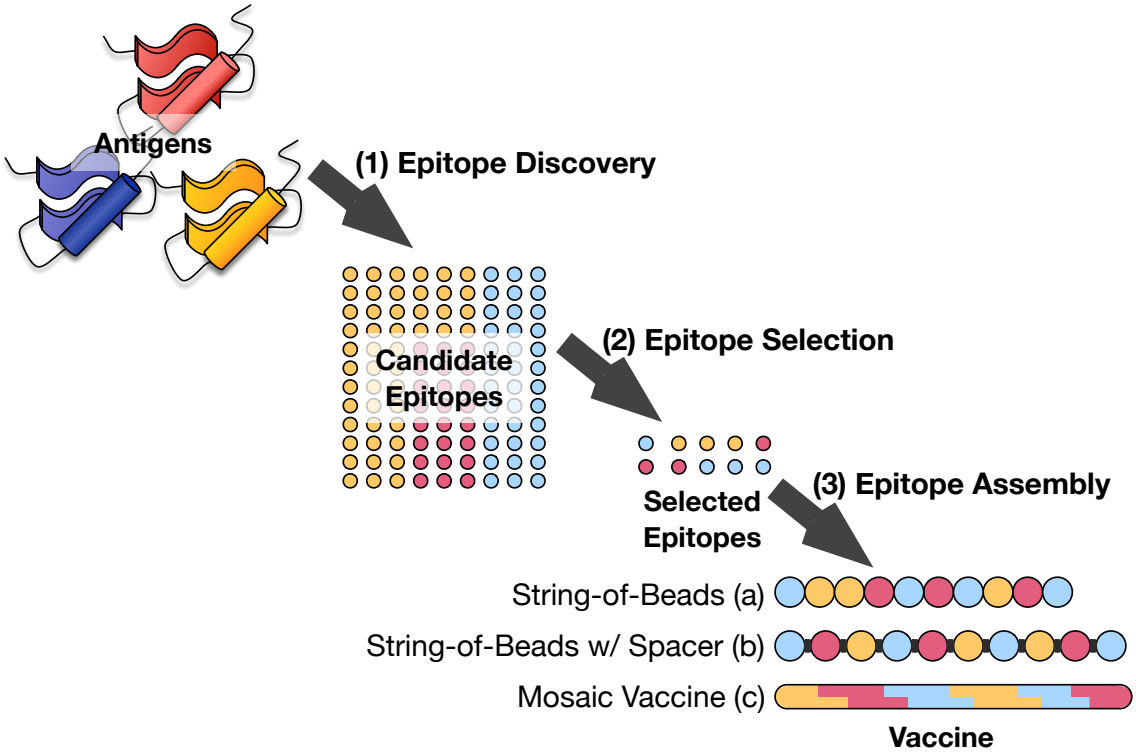


Figure 5.1: Schematic steps of EV design. Given a set of antigens, candidate epitopes are identified using experimental or algorithmic methods (1). Out of these candidate epitopes, a subset is selected that maximizes vaccine efficacy (2). These therapeutic epitopes are then assembled into a final vaccine (3). This can be done by concatenating the peptide sequences potentially using short spacer sequences to connect adjacent epitopes (3a-b), or by constructing a polypeptide of overlapping sequences (3c).

EV a perfect fit for highly personalized therapies. Moreover, EV possess many advantages over traditional vaccines in particular concerning safety, manufacturing, quality control, and storage[96].

The rational development of EVs can be divided into three steps (Figure 5.1): (1) epitope discovery, (2) epitope selection, and (3) epitope assembly. In each step, bioinformatics plays an essential role. Supervised machine learning approaches like neural networks, support vector machines, or probabilistic approaches are used during epitope discovery to predict immunogenic peptides within an antigen to accelerate this step. The selection of epitopes is the most vital step in the design process, therefore several methods have been developed to assist the selection process (see Schubert *et al.*[97] for a review). These methods differ in their emphasis of various aspects of EVs, as this topic is still highly controversial and not well understood. Nevertheless, the assembly and delivery of EVs remains a major obstacle. Several strategies haven be explored in studies delivering the selected epitopes directly as peptide cocktails, or assembled as polypeptides[98]. One particular prominent approach

assembles the epitopes like beads on a string and is hence called string-of-beads vaccines (SBV). The efficacy of such SBVs is dependent on the correct intracellular processing such that the majority of the selected immunogenic peptides are recovered during antigen processing and subsequently presented on HLA molecules. A major factor in optimal recovery is the correct cleavage of the epitopes, which was linked to the ordering of the peptides within the SBV due to its influence on cleavage probability [14]. An unfavorable order can lead to miscleaved epitopes, which in turn reduces the efficacy of the vaccine (Figure 5.2). Even new cleavage sites and non-therapeutic neo-epitopes can arise at junctions between epitopes, which can have detrimental effects [99].

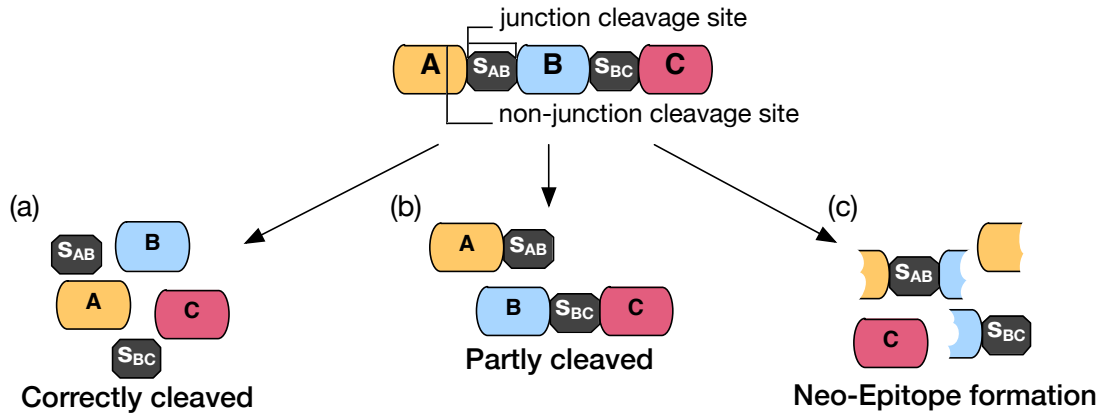


Figure 5.2: Possible cleavage outcomes of a SBV. The efficacy of a SBV depends on correct proteasomal cleavage. Desired is a cleavage pattern that correctly recovers all contained epitopes (a). Not all junction cleavage sites might be cleaved, which results in a partly cleaved and less effective SBV (b). Cleavage of the SBV at non-junction sites can create neo-epitopes. Generation of neo-epitopes can induce unwanted immune responses and reduces the amount of desired epitopes generated by the SBV (c).

Several experimental groups have proposed the use of so-called spacers, small amino acid sequences connecting two epitopes, to increase the recovery probabilities of the therapeutic epitopes [13,15,16]. However, the length and sequence of these spacers were not thoroughly optimized and often the same spacer sequence was used throughout the whole SBV due to the high experimental burden, although it was clear that the spacer sequence has to be individually determined for each epitope pair to fully exploit its potential. An experimental validation of all possible designs, however, is impossible even for relatively small SBVs. A dozen of therapeutic epitopes can be combined in about half a billion distinct SBVs. Considering additional spacer sequences of varying length and sequences would increase the possibilities many times over. Also, with increasing spacer length, the problem of inducing neo-epitopes and new cleavage sites becomes increasingly challenging. Thus, as of now, it was unclear how to determine the optimal length and amino acid sequence of a spacer connecting an epitope pair and the optimal order of the SBV.

Related Work

Only a few computational approaches have been proposed that address SBV construction. Vider-Shalit *et al.* suggested a genetic algorithm to simultaneously select and arrange therapeutic epitopes to a SBV[100]. Toussaint *et al.* embedded the epitope assembly problem into the well-known traveling salesman problem (TSP) and solved it via integer linear programming or heuristically[101]. But neither of these approaches considered spacer sequences. Only the method proposed by Antonets *et al.*[102], allowed for the construction of spacer sequences and assembled the epitope-spacer pairs into a final SBV using the TSP embedding proposed by Toussaint *et al.* However, the method uses a uninterpretable objective function as it combines multiple terms resembling predictable aspects of the HLA I antigen processing pathway by arbitrary weights. In addition, it uses exhaustive search to find the best spacer of a predefined set of spacer sequences for pairs of epitopes, and uses a genetic algorithm to solve the arising TSP problem.

Project Overview

In this work, we propose a framework to determine a provably optimal spacer sequence of fixed length for a given HLA-I restricted epitope pair. We also extend the formulation to determine the optimal spacer length and combine this approach with that of Toussaint *et al.*[101] to design an optimal SBV with flexible spacer sequences. Additionally, we account for the problem of arising neo-epitopes and cleavage sites by formulating the problem of designing a spacer sequence as multi-objective optimization problem that maximizes the recovery probability of the desired epitopes, minimizes the immunogenicity of neo-epitopes, and (optionally) minimizes the cleavage probability at non-junction sites at the same time. More formally:

***Problem Definition**: Given a set E of N epitopes $e_1, .., e_N$, we search for the optimal order of all N epitopes as well as the length and sequences of the $N-1$ spacers between the epitopes that maximizes the recovery of the epitopes while minimizing the creation of undesired neo-epitopes. An optimal spacer s_{ij} of length k connecting two epitopes e_i and e_j is defined as the sequence s_{ij} that maximizes the likelihood that epitopes e_i and e_j are cleaved at their respective junction cleavage sites c_i and c_j. This in turn potentially increases the likelihood of recovering the therapeutic epitopes and thus the likelihood of being loaded and presented on HLA-I molecules. In the case of neo-epitope formation spanning the connected epitope-spacer pair $e_i s_{ij} e_j$, s_{ij} should be additionally designed to reduce the immunogenicity of the neo-epitopes and thus to reduce potential adverse effects.*

To quantify the neo-epitope immunogenicity and cleavage likelihood of the epitopes, standard machine learning methods have to be integrated into the optimization process.

Since computational prediction methods for proteasomal cleavage and HLA-I binding are well-established, we focus our efforts solely on HLA-I antigen processing. As the spacer sequence is defined over a discrete alphabet, namely that of all naturally occurring amino acids Σ, and, as the ordering of the epitopes is also finite, we deal with a combinatorial optimization problem. Depending on the chosen prediction models for cleavage likelihood and immunogenicity, the whole problem can become impractical to solve even for small instances (for non-linear, convex prediction models) or its underlying decision problem becomes undecidable (in the case of non-linear, non-convex prediction methods) as these models would lead to non-linear, (non-convex), mixed integer and constraint optimization problems[103,104]. Therefore, we restrict the prediction models used within the optimization to be linear.

5.2 Methods

5.2.1 Spacer Design as a Multiobjective Optimization Problem

As we could see from the discussion before, multiple design goals have to be considered when designing spacer sequences for a SBV. On the one hand, the spacer sequence s_{ij} that connects the epitope pair e_i and e_j should be designed to maximize the cleavage probability $C(\cdot)$ of the two. On the other hand, s_{ij} should be designed to decrease the potential harmful effects of arising neo-epitopes by reducing the neo-immunogenicity $I(\cdot)$ of the complete concatenated sequence $S := e_i s_{ij} e_j$. This naturally leads us to multiobjective optimization. But solving a multiobjective optimization problem can be difficult, especially in the case of discrete problems as discussed in Section 3.2. Since increasing the cleavage likelihood and thus increasing the efficacy of the SBV is clearly more important than to reduce potential number of neo-epitopes by decreasing the neo-immunogenicity, the stated problem exhibits a clear priority in its objectives. Hence, finding a Pareto-optimal solution can be drastically simplified by applying lexicographic optimization (LO)[39]. In LO, the objectives are ranked based on their importance for the designer and several single objective problems of the form

$$\begin{aligned} &\min_x z_i(x) \\ &\textbf{s.t. } z_j(x) \leq z_j(\hat{x}) \\ &\text{where } i \in \{1, N\}, j \in \{1, i-1\} \text{ if } i > 1, \end{aligned} \tag{5.1}$$

can be solved to find a lexicographically optimal solution. Here, i represents the priority of the objective function and $z_j(\hat{x})$ the optimum of the j-th objective function found at the j-th iteration.

5.2.2 Cleavage Site Model

We define the cleavage objective of spacer s_{ij} and epitope pair e_i,e_j as the linear combination of the individual cleavage likelihoods of cleavage site c_i and c_j predicted by a linear cleavage site model $\phi_C : \Sigma \times \mathbb{N}_0 \to \mathbb{R}$:

$$C(e_i, e_j | s_{ij}) := \sum_{l=0}^{n_c-1} \phi_c(S[i_c + l], l) + \phi_c(S[j_c + l], l). \tag{5.2}$$

Here $S := e_i s_{ij} e_j$ denotes the concatenated sequence of a spacer and its enclosing epitope pair, $S[x]$ indicates the x-th character of sequence S, n_c represents the number of amino acids used to predict a cleavage site, and i_c,j_c denote the start of the segments used to predict the cleavage likelihood at site c_i and c_j, respectively. The value of $\phi_c(a, i)$ of amino acid a at position i represents the influence of that amino acid at a position i on the cleavage log-likelihood. Thus the log-likelihood is obtained by summing over the values of $\phi_c(X[i], i)$ for a given amino acid sequence X of length n_c.

5.2.3 Immunogenicity Model

The immunogenicity objective is based on the formulation proposed by Toussaint *et al.*[105], which was used in the context of epitope selection. The formulation assumes that each epitope contributes independently to the overall immunogenicity with respect to a target population or an individual represented by a set of HLA alleles H. The impact of each HLA allele is directly proportional to its probability p_h of occurring in any individual of the target population. In the case of a personalized setting, p_h could be substituted with normalized relative expression of the respective HLA allele, or simply equally weighted.

Since immunogenicity prediction of an epitope is still an unsolved problem, binding affinity is often used to approximate the immunogenicity of a single peptide as there is a strong correlation between those two properties[106]. However, peptides with an insufficient binding affinity are considered as non-binders, and thus should not contribute to the (neo)-immunogenicity of a SBV construct. Hence, it is only necessary to alter the spacer sequence s_{ij} connecting the epitope pair e_i, e_j if the artificial peptides of length n_e spanning the two epitopes and the connecting spacer exceed a certain binding threshold τ_h necessary to bind to HLA allele h. To account for that, only artificial peptides above the defined binding threshold enter the objective function, while non-binding peptides do not influence

the objective. We thus arrive at the immunogenictiy objective I:

$$I(S|H) := \sum_{h \in H} p_h \sum_{i=1}^{n-n_e} \max(0, \sum_{j=0}^{n_e-1} \phi_I(S[i+j], h, j) - \tau_h), \tag{5.3}$$

where S is the input sequence of length n, and $\phi_I : \Sigma \times H \times \mathbb{N}_0 \to \mathbb{R}_{\geq 0}$ represents a linear binding affinity prediction model for an HLA allele $h \in H$.

5.2.4 Spacer Design with Fixed Length

We first formulate the problem of designing a spacer of fixed length k as a bi-objective mixed integer linear program (BOMIP) and cast it into its LO representation. We represent each position in $S := e_i s_{ij} e_j$ as a set of amino acids. The sets describing e_i and e_j only contain the amino acids appearing in e_i and e_j: $S_l := \{S[l]\}$ for $l \in [0, .., |e_i|, |e_i|+k, ..., |e_i|+k+|e_j|]$. The sets defining the spacer sequence s_{ij} can contain all naturally occurring amino acids: $S_l := \Sigma$ for $l \in [|e_i|, |e_i| + k - 1]$. Additionally, each position i and amino acid $a \in S_i$ is assigned a binary decision variable $x_{i,a}$ taking on $x_{i,a} = 1$ iff amino acid a at position i is chosen in the final design. To ensure correct peptide sequences, a constraint has to be added allowing only one amino acid per position. The complete bi-objective mixed integer problem thus becomes:

$$\begin{aligned}
&\text{(O1)} \quad \max_{\mathbf{x}} \sum_{l=0}^{n_c-1} \Big(\sum_{a \in S_{i_c+l}} x_{i_c+l,a} \phi_c(a, l) + \sum_{b \in S_{j_c+l}} x_{j_c+l,b} \phi_c(b, l) \Big) \\
&\text{(O2)} \quad \min_{\mathbf{x}} \sum_{h \in H} p_h \sum_{i=1}^{n-n_e} \max(0, \Big(\sum_{j=0}^{n_e-1} \sum_{a \in S_{i+j}} x_{i+j,a} \phi_I(a, h, j) \Big) - \tau_h) \\
&\textbf{s.t.} \\
&\text{(C1)} \quad \forall\, i \in \{1, n\} \qquad \sum_{a \in S_i} x_{i,a} \leq 1
\end{aligned} \tag{5.4}$$

To obtain a lexicographically optimal solution of this BOMIP, we solve two consecutive MIPs:

$$\mathrm{LO}_{\mathrm{spacer}}(e_i, e_j, k) := \tag{5.5}$$

$$\text{P1:} \quad \hat{z}_1 := \max_{\mathbf{x}} \sum_{l=0}^{n_c-1} \Big(\sum_{a \in S_{i_c+l}} x_{i_c+l,a} \phi_c(a,l) + \sum_{b \in S_{j_c+l}} x_{j_c+l,b} \phi_c(b,l) \Big)$$

s.t.

$$\forall\, i \in \{1,n\} \qquad \sum_{a \in S_i} x_{i,a} \leq 1$$

$$\text{P2:} \quad \hat{z}_2 := \min_{\mathbf{x}} \sum_{h \in H} p_h \sum_{i=1}^{n-n_e} \max\Big(0, \Big(\sum_{j=0}^{n_e-1} \sum_{a \in S_{i+j}} x_{i+j,a} \phi_I(a,h,j)\Big) - \tau_h\Big)$$

s.t.

$$\forall\, i \in \{1,n\} \qquad \sum_{a \in S_i} x_{i,a} \leq 1$$

$$\sum_{l=0}^{n_c-1} \Big(\sum_{a \in S_{i_c+l}} x_{i_c+l,a} \phi_c(a,l) + \sum_{b \in S_{j_c+l}} x_{j_c+l,b} \phi_c(b,l) \Big) \geq \alpha \hat{z}_1$$

Here, we restrict P2 to obtain at least $\alpha \in [0,1]$ fraction of the maximal cleavage score achieved by solving P1. α represents the trade-off between cleavage likelihood and the likelihood of decreasing the immunogenicity score.

5.2.5 Non-junction Cleavage Site Minimization

As to further reduce side effects and increase efficacy, non-junction cleavage sites arising due to the introduced spacer sequences and order of the epitopes within the SBV, should be kept at a minimum. This can be achieved by additionally minimizing the non-junctional cleavage likelihood, which we define as the linear combination of the predicted cleavage likelihoods ϕ_c of all non-junctional cleavage sites. Such an additional design goal can easily be incorporated into the already existing framework by adding a third optimization problem

to the sequences of consecutively solved MILPs:

$$\mathrm{LO}_{\mathrm{spacerEx}}(e_i, e_j, k) := ... \tag{5.6}$$

$$\text{P3} \quad \hat{z}_3 := \mathbf{min_x} \sum_{i=1;i\neq i_c;i\neq j_c}^{n-n_c} \sum_{j=0}^{n_c-1} \sum_{a\in S_{i+j}} x_{i+j,a}\phi_c(a,j) \tag{5.7}$$

s.t.

$$\forall\ i \in \{1,n\} \quad \sum_{a\in S_i} x_{i,a} \leq 1$$

$$\sum_{l=0}^{n_c-1}\Big(\sum_{a\in S_{i_c+l}} x_{i_c+l,a}\phi_c(a,l) + \sum_{b\in S_{j_c+l}} x_{j_c+l,b}\phi_c(b,l)\Big) \geq \alpha\hat{z}_1$$

$$\sum_{h\in H} p_h \sum_{i=1}^{n-n_e} \max\Big(0, \Big(\sum_{j=0}^{n_e-1}\sum_{a\in S_{i+j}} x_{i+j,a}\phi_I(a,h,j)\Big) - \tau_h\Big) \leq (2-\beta)\hat{z}_2$$

Here again, $\alpha \in [0,1]$ and $\beta \in [0,1]$ represent the trade-offs between the three objective functions.

5.2.6 String-of-Beads Design with Spacers of Flexible Length

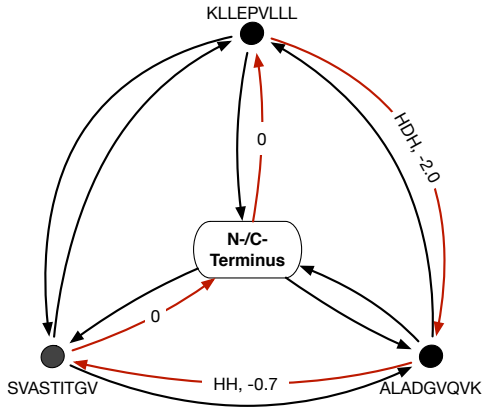
To design SBVs with spacer sequences of flexible length, the LO formulation is iteratively solved for each epitope pair and varying spacer length $k \in [0,..,K]$. The spacer design with the highest minimum of both cleavage site likelihoods is selected for each epitope pair (Algorithm 5.3a).

To find the best orientation, a fully connected and directed graph is initialized, where each node represents an epitope, and each edge represents the best spacer connecting the two epitopes. The edge weights are assigned to the negative cleavage log-likelihood of the corresponding spacer-epitope pair. Following Toussaint *et al.*, a TSP instance is formulated based on this graph by adding a node that represents the N- and C-terminus of the SBV and connecting it with zero edge weights to all other nodes (Figure 5.3b). Solving this formulated TSP instance yields an optimal ordering of the epitopes. Together with the optimized spacers we thus obtain an optimal sequence for the entire vaccine construct.

This TPS instance can be solved using ILP techniques. Many different compact and non-compact ILP formulations of the TSP exist varying in their tightness of the optimality-gap (surveyed in Orman *et al.*[107]). However, runtime can still be impracticable even for small instances. That is why we use the TSP heuristic proposed by Lin and Kernighan[108], which was later refined by Helsgaun[109]. The Lin-Kernighan heuristic (LKH-2) is a local search algorithm using the k-opt move as main search routine. A k-opt move changes a tour by replacing k edges from the tour with k new edges such that the resulting tour is

```
input  : E set of epitopes
input  : K max. spacer length
output : Optimal spacer s_ij for each epitope pair
         e_i, e_j ∈ E
1  begin
2      A ← −∞ (a matrix of size E × E)
3      S ← ∅ (an array holding spacers s_ij)
4      //For-Loops are executed in parallel
5      foreach e_i, e_j ∈ E × E do
6          for k ← 0 to K do
7              s_ij ← LO_spacer(e_i, e_j, k)
8              if min(C(e_i|s_ij), C(e_j|s_ij)) > A[i, j]
                then
9                  A[i, j] ←
                     min(C(e_i|s_ij), C(e_j|s_ij))
10                 S[i, j] ← s_ij
11             end
12         end
13     end
14 end
```

(a) Spacer design of flexible length



(b) TSP graph for SBV design

Figure 5.3: (a) The algorithm for designing spacer sequences of flexible length for each epitope-pair. (b) Based on the optimal spacer sequences and length a fully connected and directed graph is generated, where each node represents an epitope and each edge represents the determined spacer with its negative cleavage likelihood. By adding a dummy node and connecting it with all other nodes, a traveling salesman instance can be formulated, which determines the optimal ordering of the string-of-beads vaccine that maximizes the overall epitope recovery.

shorter. Because the number of moves increases exponentially with k, k is usually restricted to $k = 2$ or $k = 3$. The used k-opt move in LKH-2 however is an exception. Instead of *a prior* specifying k, LKH-2 is adaptive and selects the k that leads to a shorter tour. The runtime of LKH-2 is approximately $O(n^{2.2})$ [110].

5.2.7 Implementation

The framework was implemented in Python 2.7 and fully integrated into Fred 2 (Section 7.2). To efficiently solve the LO formulation we employed CPLEX 12.6 together with Pyomo 4.2. For the epitope ordering we used the LKH-2 implementation of Helsgaun [109]. The implementation supports *SYFPEITHI* [111], *SMM* [112], *SMMPBMC*, and *BIMAS* [113] as internal linear binding affinity prediction model, and *PCM* [114], and *ProteaSMM* [115] as internal cleavage site prediction model. The source code is published under a 3-clause BSD license and can be found at `https://github.com/FRED-2/OptiVac`.

5.3 Results

For the purpose of this study *SYFPEITHI* [111] and *PCM* [114] have been used if not stated otherwise. Statistical analysis was performed with R (`www.r-project.org`). Statistical significance was considered at a significance level of 0.05.

A pool of nine-mer epitopes was predicted for proteins of the cytomegalic virus strain AD169 (UniProt Proteom ID: UP000008991). A peptide was considered an epitope if it exceeded a predicted SYFPEITHI-Score of 20 for at least one HLA allele prevalent in the European population. Based on this epitope pool, several experiments were conducted to validate the model performance.

5.3.1 Evaluation of *in silico* Designed Spacers

One thousand random epitope pairs, were generated and spacers of length 1-6 designed, optimized for the HLA distribution of the European population using $\alpha = 0.99$. Fold-changes in cleavage likelihood as well as neo-immunogenicity were compared with concatenated epitopes without spacers, a commonly used fixed spacer (AAY) [15,116,117], and with optimally determined spacers (Figure 5.4).

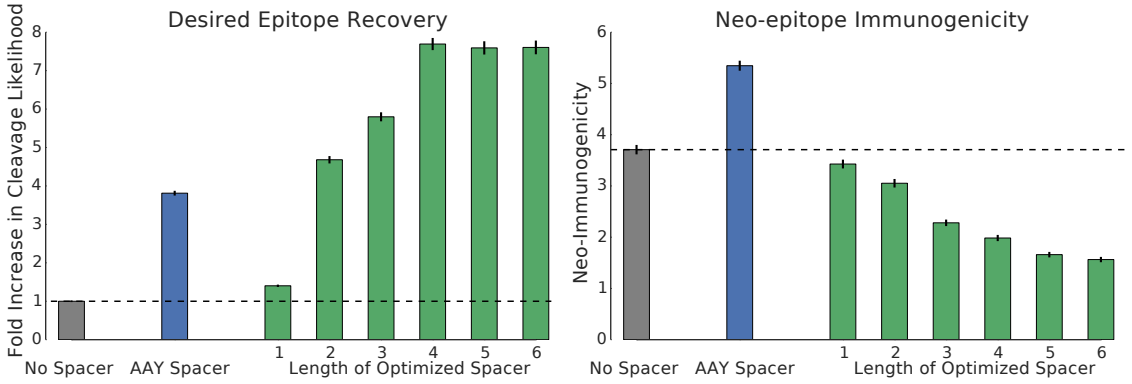


Figure 5.4: Fold change in cleavage likelihood and differences in neo-immunogenicity compared for 1,000 randomly sampled epitope pairs. Spacers of lengths 1 - 6 were designed with the described model. The cleavage probability (left) and immunogenicity (right) were compared for epitope pairs concatenated without a spacer sequence, epitope pairs combined with a commonly used spacer sequence (AAY), and pairs combined with optimally designed spacers. Black error bars represent the 68% confidence intervals.

At each spacer length, a significant increase in cleavage likelihood could be observed for epitope pairs with optimized spacers compared to epitope pairs without spacers (paired one-sided Wilcoxon signed-rank test, Bonferroni-corrected). Also, the optimized spacers outperformed the constructs with fixed spacer after a length of two (paired one-sided Wilcoxon signed-rank test, Bonferroni-corrected). Maximal increase in cleavage likelihood

was already achieved with a spacer length of four, which is not surprising since the applied cleavage model uses four C- and two N-terminal amino acids to predict a cleavage site. The use of optimal spacer sequences resulted in a 7.7-fold increase in cleavage likelihood compared to epitope pairs without spacer sequences and a two-fold increase compared to epitope pairs with a fixed AAY-spacer. Also, significant improvements could be observed in terms of reduced neo-immunogenicity when using optimized spacers compared to both designs with fixed spacers and without spacers (paired one-sided Wilcoxon signed-rank test, Bonferroni-corrected). With increasing spacer length, the immunogenicity decreased when using optimal spacer sequences. An average neo-immunogenicity reduction of 1.9-fold and 2.7-fold could be achieved at a spacer length of four compared to epitope pairs without spacers and fixed spacers respectively.

5.3.2 Evaluation of String-of-beads Designs with Optimal Spacers

Out of the pool of epitopes, random sets of size $l = \{3, 5, 10, 15, 20, 25, 30\}$ were selected and the optimal ordering was determined for the string-of-beads construct without (SBV) and with spacer sequences ($\text{SBV}_{\text{spacer}}$) for a maximal spacer length of $k = 6$ amino acids. Additionally, ten randomly ordered string-of-beads with fixed AAY spacers (SBV_{AAY}) for the given epitope set were generated. This procedure was repeated fifty times for each set size and the junction cleavage likelihood averaged over the number of arising junction sites, the fraction of recovered epitopes (i.e., epitopes with preceding and succeeding C-terminal cleavage site scores with positive cleavage score), as well as the neo-immunogenicity of the complete construct normalized by the number of included epitopes were compared between the string-of-beads with spacer, without spacer sequences, and the average performance of the random constructs with fixed spacers (Figure 5.5).

The average junction cleavage scores of $\text{SBV}_{\text{spacer}}$ and SBV_{AAY} were stable and well above the cleavage threshold of zero for all set sizes with an average score of 1.74 ± 0.63 and 0.73 ± 0.53 respectively, whereas the average junction cleavage score of SBV decreased with increasing set sizes and was below the cleavage threshold even for small set sizes with an average score of -0.85 ± 1.09. This was also reflected in the percentage of recovered epitopes. SBV exhibited a decreasing recovery with increasing set sizes with an average of $15.4 \pm 24.3\%$, while $\text{SBV}_{\text{spacer}}$ and SBV_{AAY} achieved a stable average recovery of $78.3 \pm 16.2\%$ and $62.7 \pm 15.2\%$, corresponding to a 5-fold and 4-fold increase, respectively. $\text{SBV}_{\text{spacer}}$ also consistently outperformed SBV_{AAY} both in cleavage likelihood (2.38-fold increase) and recovery rate (1.25-fold increase). The differences in neo-immunogenicity were not as pronounced, which was expected due to the chosen value of α. $\text{SBV}_{\text{spacer}}$ consistently achieved a lower neo-immunogenicity score (average 1.88 ± 0.59) than SBV

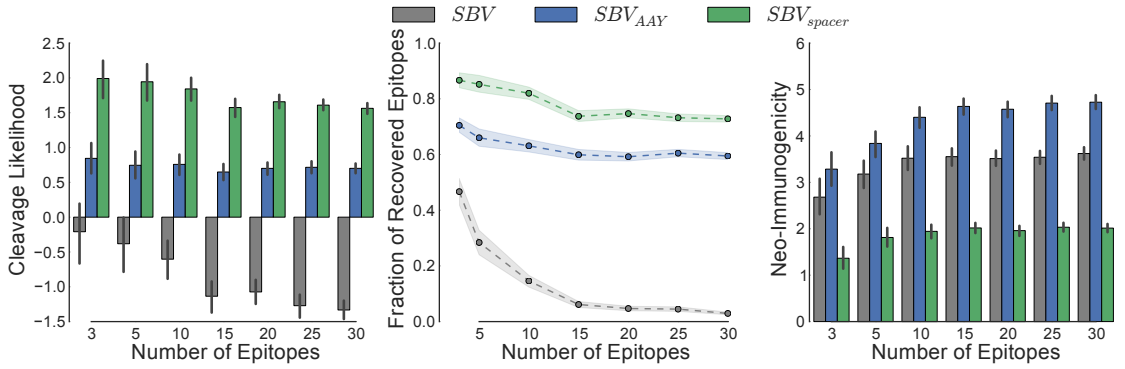


Figure 5.5: Comparison of string-of-beads with and without spacer sequences. Average junction cleavage likelihood (a), recovery percentage (b), and neo-immunogenicity (c) were measured for optimal string-of-beads designs with, without, and fixed AAY spacers. The string-of-beads constructs comprised 3 to 30 randomly selected epitopes. For each set size, the sampling was repeated 50 times. The maximum spacer length was set to $k = 6$. Black error bars and colored outlines represent the 68% confidence intervals.

(average 3.37 ± 0.93) and SBV_{AAY} (average 4.31 ± 0.99) resulting in a decrease of 44.2% and 56.8%, respectively.

The optimal spacer length averaged at 3.23 ± 0.50 AA. The runtime for instances with 30 epitopes was 5 min on average on current commodity hardware (12-core Intel Xeon E5-2620 running at 2 GHz).

5.3.3 Comparison of Experimentally used Designs with Optimized Designs

Several spacer sequences have been proposed in various settings ranging from prophylactic vaccine to therapeutic cancer vaccine studies[13,15,116,118–120]. However, these spacer sequences are not universally applicable and their effectiveness depends on the epitope pairs they connect. To show the potential efficacy of the proposed model, we compared multiepitope studies that used spacers with our *in silico* designed spacers in terms of epitope recovery and induced neo-epitopes. An epitope was considered recovered if its preceding and succeeding cleavage sites were likely to be cleaved, as predicted by PCM[114] (i.e., PCM-score > 0.0). Neo-epitope prediction was performed with SYFPEITHI[111] using the default threshold (i.e., SYFPEITHI-Score ≥ 20).

Levy *et al.* proposed a therapeutic multiepitope polypeptide consisting of HLA-A*02:01 restricted modified epitopes derived from different melanoma associated antigens (gp100:209-217(210M): IMDQVPFSV, gp100:280-288(288V): YLEPGEVTV; Mart1:27-35(27L): LAGIGILTV; tyrosinase: 368-376(370D): YMDGTMSQV) and showed the proteasomal dependent efficacy in vitro using PBMC of healthy donors and patients undergoing treatment[116]. To

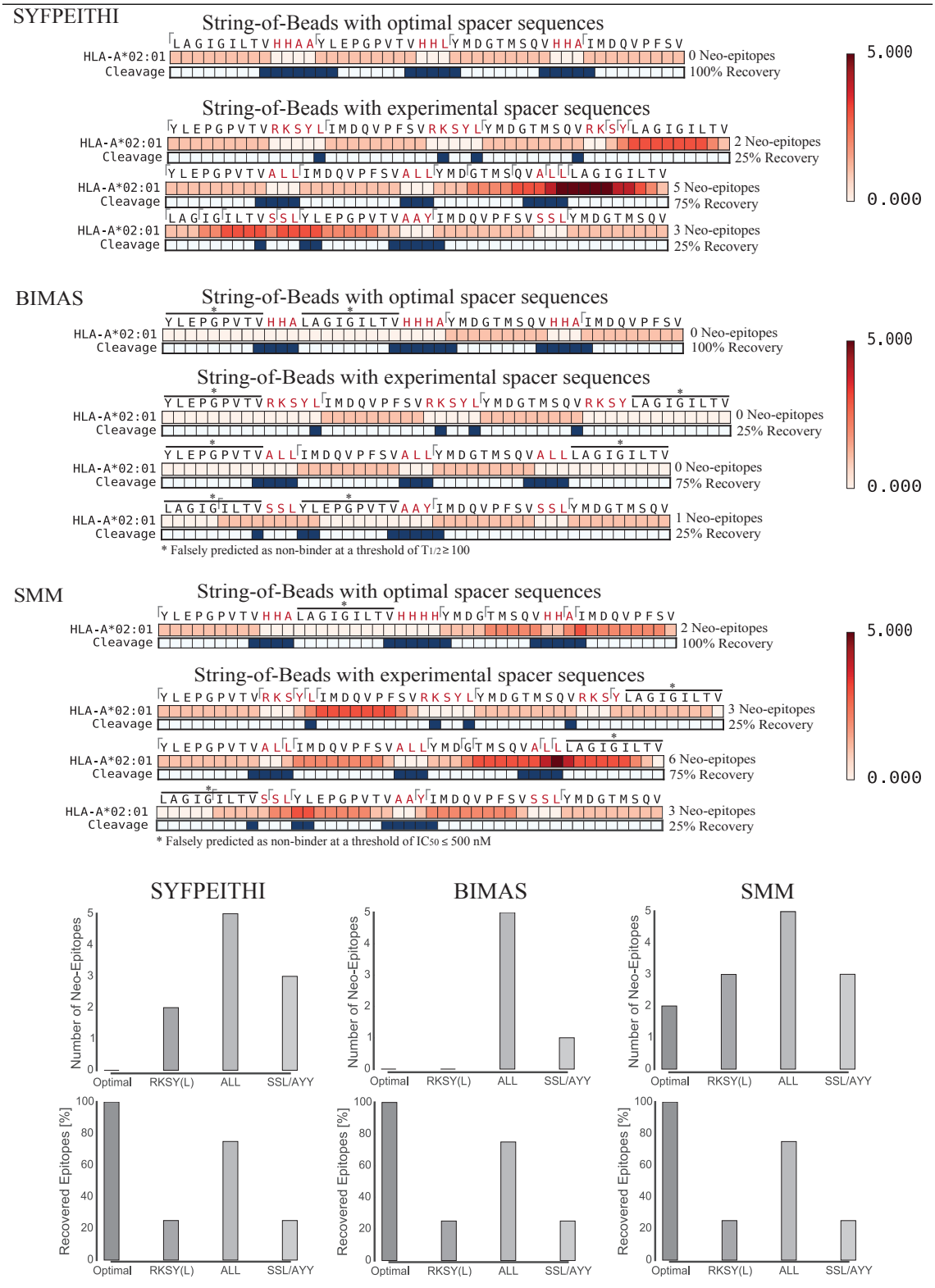


Figure 5.6: Spacer sequences were constructed with SYFPEITHI[111], BIMAS[113], and SMM[112]. Cleavage prediction was performed with PCM[114], classifying a site as cleaved if its score was greater than zero. The epitope thresholds used for neo-epitope detection were SYFPETHI-score ≥ 20, BIMAS ≥ 100 $T_{1/2}$, and SMM ≤ 500 nM. Red bars represent predicted epitopes and the intensity indicates overlapping epitopes at that position. The blue rectangles represent predicted C-terminal cleavage sites. Spacer sequences are marked in red. A tick indicates the start position of a predicted nine-mer epitope. Although the different prediction methods yielded different spacer sequences, the overall result remained the same. The in silico designed spacers were superior in terms of recovered epitopes and neo-epitope formation.

combine the selected peptides, naturally derived spacer sequence (RKSY(L)) as well as experimentally derived spacers (AAY, ALL/SSL) were used. The selected epitopes were included multiple times in the polypeptide combined with the different spacers to maximize the recovery probability. Therefore, we compared the different segments of the vaccine that were connected with the same spacer sequences (Figure 5.6).

In general, the optimal SBV design outperformed the experimentally used spacer sequences both in terms of therapeutic epitope recovery and in reduced neo-epitope appearance. With the designed spacers, 100% of therapeutic epitopes could be recovered without generating neo-epitopes spanning the spacer sequences. The experimentally used spacers on the other hand either generated neo-epitopes or were not able to recover an essential amount of the therapeutic epitopes. With the spacer RKSY(L), only one out of four epitopes could be recovered. ALL induced five neo-epitopes spanning the spacer and the Mart1 derived epitope, while the combination of SLL and AAY generated neo-epitopes and did result in recovery of one out of four epitopes only. Even the design with optimally ordered epitopes and selected experimental spacer sequences could not recover all epitopes and introduced neo-epitopes. In order to establish the effect of different (linear) epitope prediction methods, the comparison was repeated with different methods (BIMAS[113], SMM[112]). The recovery analysis was again performed with PCM, and default thresholds for BIMAS (predicted $T_{1/2} \geq 100$) and SMM (predicted $IC_{50} \leq 500nM$) were used for neo-epitope detection. All therapeutic epitopes could be recovered using the *in silico* designed spacers with a smaller or equal number of neo-epitope compared to the best experimentally used spacer sequence. While there are differences in detail between the methods, their overall behavior remained the same (Figure 5.6). Differences can be attributed to variation in prediction accuracy of the methods.

Similar results could be observed for the proposed SBV construct of Ding *et al.*[13] (Appendix Figure E.1). The proposed SBV was composed of HBV X protein derived T-cell epitopes, which were combined with different spacer sequences to reduce the number of junction neo-epitopes. With the *in silico* designed spacer sequences, all therapeutic epitopes could be recovered without introducing neo-epitopes, whereas the experimentally used spacers induced neo-epitopes and were not able to recover all therapeutic epitopes.

5.4 Discussion

In this work we proposed a mathematical model for designing spacer sequences of flexible length for string-of-beads vaccines by exploiting existing proteasomal cleavage and epitope prediction methods, and combined the model with a TSP approach for optimal epitope ordering. We also addressed the problem of neo-epitopes and non-junction cleavage sites arising by spacer sequences and order of epitopes within the string-of-beads by extending the

formulation with two additional objective functions. To efficiently solve the multi-objective optimization problem we employ lexicographical optimization techniques.

The efficacy of the model was shown by comparing the recovery rates and neo-immunogenicity of optimal designs with commonly used fixed spacer sequences and spacer-less designs. The optimal design led in each case to increased predicted epitope recovery and reduced generation of neo-antigens. We also compared experimentally tested string-of-beads designs that used spacer sequences with our optimized designs. The experimentally used spacer sequences were often sub-optimally chosen for the connecting epitopes. As a consequence, neo-epitopes spanning the spacer sequences arose or proteasomal cleavage could not be guided to cleave the therapeutic epitopes correctly. In contrast, the *in silico* designed string-of-beads with optimally determined spacers showed improved cleavage patterns and reduced neo-immunogenicity. Often, all therapeutic epitopes could be correctly cleaved without introducing neo-epitopes.

An obvious limitation of the current method is its reliance on computational models for proteasomal cleavage and epitope prediction. While models for HLA class I binding prediction exhibit a high accuracy, proteasomal cleavage models still leave room for improvements[121]. Currently, the approach is restricted to HLA class I epitopes but could be effortlessly extended to HLA class II epitopes once a cleavage prediction method for HLA-II ligands becomes available. Also, the framework is designed flexibly enough to replace the underlying proteasomal cleavage prediction method, once more reliable computational prediction models are published.

An experimental validation of selected optimal spacer designs is a non-trivial task. It cannot be performed as exhaustively as our computational study - the mere number of possible designs is simply too large. An experimental validation will thus most likely be limited to comparing only a few selected optimal designs to fixed-spacer or spacer-less designs.

In conclusion, our method is the first framework that optimally designs both epitope order and spacers for a string-of-beads vaccine design. The mathematical method employs state-of-the-art prediction methods, but does not depend on specific methods. Our model predicts an increased recovery of desired epitopes and a reduced production of neo-epitopes compared to both fixed-spacer and spacer-less designs.

Chapter 6

De-immunization of Biotherapeutics

The content of this chapter is part of an unpublished manuscript:

Schubert, B., Schärfe, C.S., Dönnes, P.,* et al. *(2016).*
De-immunization of Factor VIII using the evolutionary Hamiltonian.

6.1 Introduction

Biopharmaceutical drugs, produced by recombinant expression, have become the third pillar of the pharmaceutical industry, besides chemically synthesized small molecule drugs and natural products[122]. Biopharmaceutical drugs, also called biotherapeutics, are biomolecules that display specific pharmacological activities that can be harnessed for therapeutic use or diagnostic purposes. The majority of biotherapeutics are protein drugs such as cytokines, hormones, growth factors, enzymes, and antibodies. But also, DNA and RNA therapies, as well as vaccines belong to the category of biopharmaceutical drugs. Since the development of recombinant insulin in the 1980s, the market of biotherapeutics has drastically increased and reached a cumulative market value of $140 billion in 2013[123]. Also, the number of approved biotherapeutics remains steady with ~ 55 new products annually since the mid-1990s, representing roughly 26% of all genuinely new drugs (averaged over a four year period starting 1995)[123].

Although biotherapeutics show high activity and specificity and often are the only available treatment, a critical drawback of their use is the buildup of the patient immune response over time[17]. This immune response involves the formation of anti-drug antibodies (ADA) to these biotherapeutics and can cause reduced efficacy up to complete loss of effect or even adverse reactions[17,122,124]. ADA formation is an inherent problem of all classes of

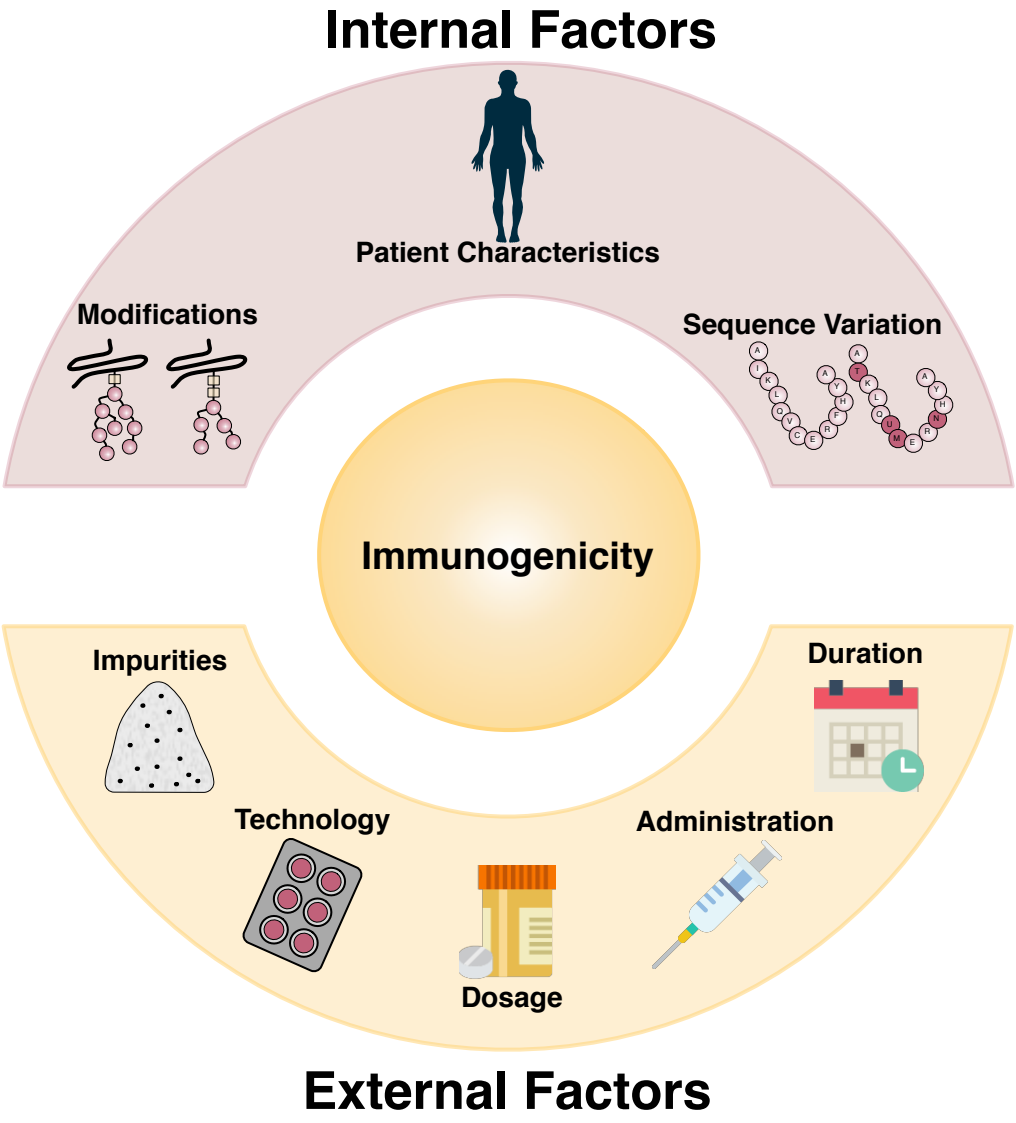


Figure 6.1: Intrinsic and extrinsic factors influencing the immunogenicity of a biotherapeutic. Besides extrinsic factors such as impurities, dosage, and duration, as well as the route of administration, intrinsic factors specific to the biotherapeutic and the patient using the drug influence the immunogenicity of the protein and thus the likleihood of anti-drug antibody formation. Icons designed by Freepik and Madebyoliver from Flaticon.

biotherapeutics and is caused by multiple factors that can be distinguished by intrinsic and extrinsic factors (Figure 6.1).

Manufacturing factors such as the host cell system that is used for recombinant protein production, contaminants and process-related impurities, as well as the type of storage, have an influence on immunogenicity of the biotherapeutic early on[17]. But also, the treatment factors such as route of administration, as well as the dosage and duration are extrinsic factors contributing to the immunogenicity of the biotherapeutic. Several studies have shown that an intramuscular and subcutaneous administration route is correlated with high immunogenicity, whereas an intravenous route is comparably less immunogenetic[125,126]. Chronic administration and high dosages are also accompanied by a high risk of developing an anti-drug immune reaction[17]. Even though, it seems that dosage and duration do not have a cumulative effect[127].

Intrinsic factors include sequence variations, genetic factors of the patient, and post-translational modification of the biotherapeutic. Not surprisingly, biotherapeutics derived from non-human sources (e.g., streptokinase[128], salmon calcitonin[129], bovine adenosine deaminase[130]) are more immunogenic than human-derived therapeutics. Although, even biotherapeutics of human origin can elicit strong, clinically relevant immune reactions[131–134]. Also, patients that suffer from a deficient gene and are treated with a replacement therapy such as factor VIII in the case of hemophilia A are more prone to develop ADAs since they might lack the tolerance healthy individuals with a functioning gene would have[135]. In some cases, the adverse immune reaction of the biotherapeutic is related to a specific HLA genotype[136,137]. Post-translation modification of the biotherapeutic (or the lack thereof) also has been associated with the immunogenicity of several biotherapeutics. For example, the high immunogenicity of the nonglycosylated, bacteria-derived form of granulocyte macrophage colony stimulation factor (GM-CSF) is caused by the exposure of antigenic sites[138]. While hyper-glycosylation has not been found to be correlated with increased immunogenicity so far[139], oligosaccharide side chain enrichment has become a widespread technique to increase serum half-life, solubility, and stability[140]. Another method to increase serum half-life, stability, and in some cases also to reduce immunogenicity is PEGylation. In the process of PEGylation, the biotherapeutic reacts with monomethoxy polyethylene glycol (mPEG) via a linker such as the primary amino groups -NH_2 of the N-terminus or from the lysine residues of the therapeutic protein or peptide[141]. Other commonly used functional groups are thiol (-SH), which are used for side specific pegylation in antibodies, and secondary amine groups (-NH-) in oligonucleotides, or carboxylic acid (-COOH) or hydroxyl (-OH) in small molecules[141].

Since immunogenicity of therapeutic proteins is an intrinsic problem, reduction of immunogenicity has become a major factor in drug development. The invention of immunogenicity reducing strategies for biotherapeutics was mainly driven by the development of

monoclonal antibodies (mAbs) and resulted in a process called humanization. Humanized mAbs only comprise the foreign complementarity determining regions of the variable regions, while the remainder of the structure is of human origin. In recent years, fully human mAbs have also been developed using bioengineering techniques combined with in vivo screening in mice[142]. These classes of humanized and human antibodies reduced the risk of undesirable immune reactions in antibody-based therapies[143].

Although humanization is quite effective, it is not applicable to other classes of biotherapeutics and even humanized, and fully human mAbs can still induce a clinically relevant anti-drug immune response[144]. Besides the many extrinsic factors, the cellular presentation and recognition of peptides, originating from the biotherapeutic, by the CD4+ T-cell-mediated adaptive immune system mainly affect the induction of ADAs[145] (Section 2.4). Therefore, the systematic removal of these epitopes by sequence alteration, referred to as de-immunization, has been successfully used as an alternative approach to reducing the immunogenicity of mAbs and other therapeutic proteins[18,144,146–148].

Related Work

The identification of HLA-II epitopes and introduction of immunogenicity-reducing mutations into the protein is highly complex and time-consuming. Often large screening efforts only result in very few candidate designs. Therefore, computational screening approaches have been developed to increase the success rate and decrease cost and time expense. The most simple approaches use the well-established HLA-II epitope prediction methods to predict promiscuous epitopes and reduce their predicted immunogenicity by locally altering the protein sequence[149,150]. But, the introduced mutations can have a significant impact on the protein's stability and function. Hence, such naïve approaches will inevitably produce designs impacting the biotherapeutic function. A more advanced approach proposed by Cantor *et al.* therefore combined experimental and *in silico* methods to preserve protein function[151]. First, HLA-II epitope clusters are identified *in silico* and a side-directed mutagenesis under selective conditions that ensures to retain the protein's function is applied. The so created pool of variants is tested for HLA-II binding, followed by a biochemical characterization of promising candidates. Finally, T-cell activation assays and antibody titers in transgenic mice are conducted to assess the immunogenicity of the candidate[151]. Clearly, such a protocol is still labor intensive. Therefore, advanced computation methods were developed that incorporated prediction methods, which can predict the impact of a mutation on function and stability[21]. Consequently, functionally harmful mutations were excluded from the search space[21]. But, using such a method ignores compensating effects of multiple mutations. That is why the most recent methods simultaneously minimize the protein's immunogenicity and its stability as a proxy for function by either using quantum

mechanical force fields (i.e. structural information)[152] or try to approximate the proteins fitness using sequence information[153].

Project Overview

Even the advanced approaches use a simple epitope counting objective as an approximation of immunogenicity ignoring the varying frequencies of HLA alleles in different populations. They also employ native amino acid frequency and pairwise frequencies as protein fitness estimate or rely on structural information, which is often not available. In this work we present a new formulation of the de-immunization problem that solely uses sequence information by leveraging recent advances in *ab initio* protein structure and variants effect prediction[154–158]. We also introduce a more expressive and quantitative immunogenicity objective that accounts for the HLA distributions within different populations or the HLA expression of an individual.

We define the de-immunization problem as to identify sequence alterations that remove CD4+ T-cell epitopes - and thus reduce the risk of ADA formation, without disrupting the structural and functional integrity of the protein excessively. More formally, we define the problem as follows:

***Problem Definition**: Given a protein sequence S of length n and a set M_i of possible mutation per position $1 \leq i \leq n$. Find a mutated sequence S' of S with k mutation for which $S'[i] \in M_i \;\; \forall \; 1 \leq i \leq$ holds and that minimize simultaneously:*

$$(1)\ I(S'|H)$$
$$(2)\ E(S')$$

Objective (1) describes the immunogenicity of the target biotherapeutic with H being the set of HLA alleles under consideration. Here, we use the same definition as in Chapter 5, but replace the linear HLA-I epitope binding model with an appropriate linear HLA-II epitope binding model. Objective (2), on the other and describes the overall fitness of the protein (i.e, stability and function).

6.2 Methods

6.2.1 Protein Fitness Objective

This work is based on recent developments in *ab initio* prediction of protein structure through evolutionary information contained in multiple sequence alignments (MSA) of closely related protein sequences[154,155]. Pair-wise global entropy models (also known as Potts models), allowed for the first time to accurately predict pair-wise contacts of residues

in a protein structure from sequence information alone, which in turn could be used to infer the complete structure of proteins or even of complexes[156–158]. More importantly, though, the model has been shown to be predictive of effects of mutational changes[159–161]. Here we will recap the statistical model and explain its usage in the context of biotherapeutic deimmunization (nomenclature follows Ekeberg *et al.*[162]).

In the context of statistical structure inference, a protein sequence of a multiple sequence alignment with length n is represented by a vector $\boldsymbol{\sigma} = (\sigma_1, ..\sigma_N)$, where $\sigma_i \in \{1, .., q\}$ with $q = 21$ representing the 20 amino acids and the gap sign. Hence, a MSA of size B can be represented by a matrix $\{\boldsymbol{\sigma}^b\}_{b=1}^B$. The single frequency for position $i \in 1, .., n$ and character $k \in \{1, .., q\}$ and pair-wise frequencies of positions $i, j \in 1, .., n$ and residues $k, l \in \{1, .., q\}$ can then be calculated as:

$$f_i(k) := \frac{1}{B} \sum_{b=1}^{B} \delta(\sigma_i^b, k) \tag{6.1}$$

$$f_{ij}(k, l) := \frac{1}{B} \sum_{b=1}^{B} \delta(\sigma_i^b, k)\delta(\sigma_j^b, l), \tag{6.2}$$

where $\delta(k, l)$ is the Kronecker delta. The Potts model is than the simplest statistical model $P(\boldsymbol{\sigma})$ that can reproduce $f_i(k)$ and $f_{ij}(k, l)$[162] and is defined as:

$$P(\boldsymbol{\sigma}) = \frac{1}{Z} \exp(\sum_{i=1}^{n} h_i(\sigma_i) + \sum_{1 \le i < j \le n} J_{i,j}(\sigma_i, \sigma_j)) \tag{6.3}$$

s.t.

$$P(\sigma_i = k) = \sum_{\substack{\sigma \\ \sigma_i = k}} P(\boldsymbol{\sigma}) = f_i(k)$$

$$P(\sigma_i = k, \sigma_j = l) = \sum_{\substack{\sigma \\ \sigma_i = k \\ \sigma_j = l}} P(\boldsymbol{\sigma}) = f_{i,j}(k, l),$$

where $\mathbf{w}_i$ and $\mathbf{J}_{i,j}$ are Lagrange parameters that are fitted to match the observed single-side and pair-wise residue frequencies of the MSA, and Z is the partition function. As the Potts model is highly overparameterized, regularization is used to keep the model from overfitting. In particular, an l_2-regularizer is used on $\mathbf{h}_i$ and $\mathbf{J}_{i,j}$ of the following form:

$$R_{l_2} = \lambda_h \sum_{i}^{n} \|\mathbf{h}_i\|_2^2 + \lambda_J \sum_{1 \le i < j \le n} \|\mathbf{J}_{i,j}\|_2^2, \tag{6.4}$$

where λ_h and λ_J are hyperparameters that are experimentally chosen to be $\lambda_h = 0.01$ and $\lambda_J = 0.2(n - 1)$[163]. To estimate suitable parameters for $\mathbf{h}_i$ and $\mathbf{J}_{i,j}$, the negative

log-likelihood $l(\mathbf{h}, \mathbf{J})$

$$l(\mathbf{h}, \mathbf{J}) = \log Z - \sum_{i=1}^{n} \sum_{k=1}^{q} f_i(k) h_i(k) - \sum_{1 \leq i < j \leq n} \sum_{k,l=1}^{q} f_{i,j}(k,l) J_{i,j}(k,l) \tag{6.5}$$

is minimized, penalized by the regularization term R_{l_2}. As an exact solution is unobtainable, due to the intractability of Z, instead of solving the log-likelihood of $P(\boldsymbol{\sigma})$ the pseudo-log likelihood is used as approximation[162].

As to predict interacting residues, the inferred quantities of $\mathbf{J}_{i,j}$ have to be summarized. For that, the Frobenius norm $S_{i,j}^{\mathrm{FN}}$ of the zero-gauged transformed values of $\mathbf{J}'_{i,j}$ is used, where $S_{i,j}^{\mathrm{FN}}$ is defined as:

$$S_{i,j}^{\mathrm{FN}} = \|\mathbf{J}_{i,j}\|_2 = \sqrt{\sum_{k,l=1}^{q} J_{i,j}(k,l)^2} \tag{6.6}$$

As to account for phylogenetic bias and insufficient sampling, the Frobenius norm is normalized using average-product correction introduced by Dunn *et al.*[164]:

$$S_{i,j}^{\mathrm{SC}} = S_{i,j}^{\mathrm{FN}} - \frac{S_{\cdot,j}^{\mathrm{FN}} S_{i,\cdot}^{\mathrm{FN}}}{S_{\cdot,\cdot}^{\mathrm{FN}}}, \tag{6.7}$$

where $\cdot$ corresponds to the average over the concerning position. The top-ranked coupled pairs are then used as residue-residue restrictions to fold the protein.

The presented statistical model can be seen as an approach to capture the evolutionary process of a protein family, and thus is also a description of the complex evolutionary pressure the protein family has evolved under. Since the model follows a Boltzmann-like statistic, it has been postulated[165] that the statistical energy term

$$H(\sigma_i, \sigma_j) := \sum_{i=1}^{n} h_i(\sigma_i) + \sum_{1 \leq i < j \leq n} J_{i,j}(\sigma_i, \sigma_j), \tag{6.8}$$

(or Hamiltonian) can be used to predict mutational effects on thermostability. However, the authors noted that the correlation of Hamiltonian change ΔH and ΔG might be confounded by the usually stronger functional aspect of the evolutionary pressure the protein is subjected to[165]. Recent empirical studies[159–161] showed that indeed the Hamiltonian captures not only the stability of a protein, but also functional aspects. In this sense, the Hamiltonian can be seen as a more general measure of fitness describing the complex evolutionary constraints of a protein. It is thus not surprising that the mutational change in the Hamiltonian is most predictive for phenotypes that resembled the natural evolutionary

pressure[160]. We therefore use the Hamiltonian of the inferred Potts model as the second objective function to quantify the fitness of the de-immunized biotherapeutic.

6.2.2 De-immunization Model

We solve the stated problem of de-immunization as a bi-objective mixed integer linear program (BOMILP). The model is based on Kingsford *et al.*'s ILP formulation of the side-chain placement problem[166], as its objective can be interpreted as Hamiltonian and the imposed constraints guarantee to establish the correct interactions between variables. But instead of selecting energetically favorable rotamers, we encode each state of the model as a possible amino acid variant at each position. To this end, a binary decision variable $x_{i,a}$ for each position $i \in \{1,..,n\}$ and each possible variation $a \in M_i$ is introduced with $x_{i,a} = 1$ iff the variant is part of the final mutant S'. An additional binary variable is introduced for each pair of variants and positions $w_{i,j,a,b}$ with $w_{i,j,a,b} = 1$ iff variant a at position i and variant b at position j are selected as part of the solution S'. Using x_i, the immunogenicity objective introduced in Chapter 5 can be formulated (O1). But here, $\Phi_I()$ represents a HLA-II binding model; in particular we used the linear prediction model of TEPITOPEpan[167]. TEPITOPEpan is a transfer learning approach using the original position specific scoring matrices of TEPITOPE[168] and a BLOSUM-based sequence similarities of HLA binding pockets to construct new position specific scoring matrices for HLA alleles not included in the original TEPITOPE model[167]. To be able to compare the predicted quantities of TEPITOPEpan's allele-specific models, the matrices are z-score normalized and their binding threshold adopted accordingly, as the HLA models exhibited a different score distribution.

$$(O1) \quad \min_x \sum_{h \in H} p_h \sum_{i=1}^{n-n_e} \max(0, (\sum_{j=0}^{n_e-1} \sum_{a \in S_{i+j}} x_{i+j,a} \phi_I(a,h,j)) - \tau_h) \tag{6.9}$$

$$(O2) \quad \max_x \sum_{i=1}^{n} \sum_{a \in M_i} x_{i,a} h_{i,a} + \sum_{i=1}^{n} \sum_{i<j \leq n} \sum_{a \in M_i} \sum_{b \in M_j} w_{i,j,a,b} J_{i,j,a,b}$$

s.t.

$$(C1) \quad \forall\, i \in \{1,n\} \qquad \sum_{a \in M_i} x_{i,a} \leq 1$$

$$(C2) \quad \forall\, i,a \in M_i, i > j \in \{1,..,n\} \qquad \sum_{b \in M_j} w_{i,j,a,b} = x_{i,a}$$

$$(C3) \quad \forall\, j,b \in M_i, j < i \in \{1,..,n\} \qquad \sum_{a \in M_i} w_{i,j,a,b} = x_{j,b}$$

$$(C4) \qquad \sum_{i=1}^{n} \sum_{a \in W_i} (1 - x_{i,a}) \leq k$$

The binary decision variables $x_{i,a}$ and $w_{i,j,a,b}$ are associated with their corresponding fitness terms $h_{i,a}$ and $J_{i,j,a,b}$ to form the second objective function (O2). To construct a consistent model, three constraints are introduced guaranteeing that only one amino acid per position is selected (C1) and that only pairwise interactions are considered for selected variants (i.e $w_{i,j,a,b,} = 1 \leftrightarrow x_{i,a} = 1 \wedge x_{j,b} = 1$, see C2 and C3). Constraints C2 and C3 can be further relaxed by dividing the pairwise fitness values into positive and negative sets[166], which is in practice done but disregarded here for ease of presentation. To be able to restrict the mutant to a specific number of introduced variations, constraint C4 limits the number of deviating amino acids to the wild type sequence W.

6.2.3 Pre-processing

To reduce the search space, a filtering based on position-specific amino acid frequency $f_i(a)$ (i.e., conservation) is applied. Only amino acids at position $i \in 1..n$ exceeding a certain frequency threshold τ_{freq} are considered as possible variants at this site. Hence, the set of possible variants per position is defined as $M_i := \{a \in \Sigma | f_i(a) \geq \tau_{\text{freq}}\}$. The wild type amino acid of the target protein is additionally added if it does not exceed the frequency threshold. This filtering is based on the assumption that variants, that are not or infrequently observed, are harmful due to either destabilizing effects, reduction of function, or intervening effects with interaction partners.

Other filtering methods for example based on summarized EC scores or based on prior knowledge gathered from experimental studies are imaginable. A position-wise summarized EC score $S^{\text{SC}}_{i,\cdot}$ represents the position's importance on overall fitness/structure. Hence, positions with high summarized EC score could be excluded from potential mutations to reduced stability disruption. However, this filtering scheme might restrict the optimization strongly, rendering epitope clusters impossible to remove due to too far reduced search space. Also, experimentally determined disease-linked mutations or in general deleterious mutations known prior to the de-immunization could be excluded as well. But, this could also lead to a strongly restricted optimization, as often mutational studies can only evaluate the effect of single mutations disregarding potential compensating mutations at other positions within the protein[160].

6.2.4 Solving a Bi-Objective ILP

Many real-world problems can only be adequately described by using multiple, often conflicting criteria. Such multiobjective optimization (MO) approaches can also be found in many different areas of bioinformatics. Such include phylogenetic[169], gene-, protein-, and metabolic network inference[170–172], protein and drug design[173–175], structure prediction[176,177], as well as sequence[178,179] and structure alignment[180,181]. While continuous

linear problems can be easily and exactly solved due to their convexity while combining the objectives into one with appropriate weights, discrete MOs are hard to solve, due to the existing of unsupported non-dominated points (Section 3.2.3). Often heuristics such as multiobjective evolutionary algorithms[182–184], particle swarm algorithm[185,186], or scattered search[187] have been usually applied to discrete multi-objective problems. But with the increasing efficiency of highly optimized single-objective solvers such as CPLEX or Gurobi, the research area of discrete multiobjective exact optimization has attracted attention. So called criterion-search algorithms started to emerge, leveraging the power of single-objective optimizers by reducing the multiobjective problem into multiple single objective ones that have to be solved iteratively.

However, these existing methods generally cannot utilize modern multicore systems, and lack sophisticated implementations. Even commercial solvers such as CPLEX only provides weighted-sum approaches to deal with bi-objective optimization problems, which are unusable for the purpose of discrete multiobjective optimization due to the existence of unsupported non-dominated points in such optimization problems. Recently, Boland *et al.*[49] developed a new criterion search algorithm, called balanced box approach. It applies only to bi-objective optimization problems, but Boland *et al.* could show that the balanced box approach is significantly more efficient than any other of the criterion search approaches[49]. The balanced box method is a simple divide-and-conquer approach that initially defines a search rectangle based on the optimal values of the two objectives while constraining the other and subsequently dividing the search rectangle into two smaller search rectangles in each divide-step. But again, no implementation is available of the presented method. The authors also did not exploit the obvious parallel nature of their proposed approach. Consequently, our goal was to provide a highly scalable implementation of the balanced box algorithm and efficiently exploit its parallel structure. To overcome the weak parallel nature of divide-and-conquer in the early phases of the search, we developed a two-phase approach combining techniques of the ϵ-constraint method and the balanced box method. In the first phase, we generate an approximate frontier using an evenly spaced ϵ-constraint grid. The new found non-dominated points are then used to initialize the balanced box algorithm starting in a deeper level of the divide-and-conquer search tree. In the following, we sketch the approach. For further detail on the balanced box approach, the reader is kindly referred to Boland *et al.*[49].

Preliminary Definitions

First we introduce necessary notations and concepts. A bi-objective optimization problem can be stated as follows (notation adopted from Boland *et al.*[49]): Let $\mathbf{z^1} = (z_1^1, z_2^1)$ and $\mathbf{z^2} = (z_1^2, z_2^2)$ be two points in solution space with $z_1^1 \leq z_1^2$ and $z_2^2 \leq z_2^1$. Further we

define $R(\mathbf{z^1}, \mathbf{z^2})$ to be the rectangle spanned by $\mathbf{z}^1$ and $\mathbf{z}^2$. A non-dominated point within $R(\mathbf{z}^1, \mathbf{z}^2)$ can be found with the following sequential operation (the reader is referred to Boland *et al.*[49] for the detailed proof):

$$\begin{aligned}\operatorname{lex}\min_{x\in\mathcal{X}}\{z_1(\mathbf{x}), z_2(\mathbf{x}) : & z(\mathbf{x}) \in R(\mathbf{z}^1, \mathbf{z}^2)\} := \\ (1)\quad & \bar{z}_1 = \min_{x\in\mathcal{X}} z_1(x) \\ & s.t\ z(x) \in R(\mathbf{z^1}, \mathbf{z^2}) \\ (2)\quad & \bar{z}_2 = \min_{x\in\mathcal{X}} z_2(x) \\ & s.t.\ z(x) \in R(\mathbf{z^1}, \mathbf{z^2}) \text{ and } z_1(x) \leq \bar{z}_1\end{aligned}$$

and is denoted as $\bar{\mathbf{z}} = \operatorname{lex}\min_{x\in\mathcal{X}}\{z_1(x), z_2(x) : z(x) \in R(\mathbf{z^1}, \mathbf{z^2})\}$. This operation will find a non-dominated point with smallest value of z_1 within the search rectangle $R(\mathbf{z}^1, \mathbf{z}^2)$. To find the non-dominated point with smallest value in z_2, the sequence of single-objective problems that have to be solved is reversed. When solving $\operatorname{lex}\min_{x\in\mathcal{X}}\{z_1(x), z_2(x) : z(x) \in R(\mathbf{z^1}, \mathbf{z^2})\}$, we can always assume that $\mathbf{z}^2$ is a non-dominated point and together with other specific properties of the balanced box method, one can simplify the $\operatorname{lex}\min_{x\in\mathcal{X}}$ operation to solving two single-objective optimization problems with only one additional constraint added each[49]:

$$\begin{aligned}\operatorname{lex}\min_{x\in\mathcal{X}}\{z_1(\mathbf{x}), z_2(\mathbf{x}) : & z(\mathbf{x}) \in R(\mathbf{z}^1, \mathbf{z}^2)\} := \\ (1)\quad & \bar{z}_1^1 = \min_{x\in\mathcal{X}} z_1(x) \\ & s.t\ z_2(x) \leq z_2^1 \\ (2)\quad & \bar{z}_2^1 = \min_{x\in\mathcal{X}} z_2(x) \\ & s.t.\ z_1(x) \leq \bar{z}_1^1.\end{aligned}$$

The Parallel Two-phase Balanced Box Approach

Boland's first implementation only exploited the parallel implementations of the single-objective solver used to perform the $\operatorname{lex}\min_{x\in\mathcal{X}}$ operation but not the parallel structure of the balanced box approach, thus only marginal runtime speedups could be achieved using multicore systems compared to a single threaded version of the balanced box method[49]. Here, we describe our parallel implementation of the balanced box method and certain adjustments we took to overcome the weak parallel nature of the balanced box method to increase the scalability and enable efficient use of multicore systems.

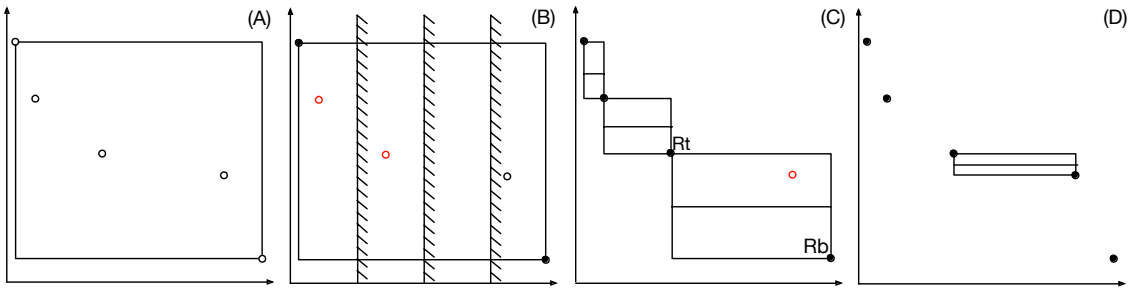


Figure 6.2: Depiction of the parallel two-phase balanced box approach. (A) first, the boundaries of the Pareto front are identified. (B) Then, the space between the boundaries is evenly divided and searched in parallel for Pareto points using the ϵ-constraint method. (C) The identified Pareto points are used to initiate rectangle search spaces which can be processed in parallel using the standard rectangle-splitting approach, by splitting the rectangle in half and searching independently the bottom and top half (D). If the corner points of the rectangles are found during the search, it is proven, that no further Pareto point resides within the search space and all points have been identified.

As a first step, the anchor points of the Pareto front are calculated by solving $\mathbf{z^T} = \operatorname{lex\,min}_{x\in\mathcal{X}}\{z_1(\mathbf{x}), z_2(\mathbf{x}) : z(\mathbf{x}) \in R((-\infty,\infty),(-\infty,\infty))\}$ and $\mathbf{z^B} = \operatorname{lex\,min}_{x\in\mathcal{X}}\{z_2(\mathbf{x}), z_1(\mathbf{x}) : z(\mathbf{x}) \in R((-\infty,\infty),(-\infty,\infty))\}$ in parallel (Figure 6.2 (A)). Then, the search space within $R(\mathbf{z^T}, \mathbf{z^B})$ is evenly constrained based on boundary conditions enforced w.l.o.g. on z_1 (Figure 6.2 (B)). The boundaries are calculated for a predefined number of constraints m with

$$\tau_i^{z_1} = z_1^T + \frac{i \cdot (z_1^B - z_1^T)}{m} \quad \text{with } 1 \leq i \leq m. \tag{6.10}$$

Each of the so defined sections of the search space can be searched independently by solving $\mathbf{z^i} = \operatorname{lex\,min}_{x\in\mathcal{X}}\{z_1(\mathbf{x}), z_2(\mathbf{x}) : z(\mathbf{x}) \in R((\tau_i^{z_1}, z_2^T), \mathbf{z^B})\}$ and the resulting new non-dominated points can be used as initial approximation of the Pareto front. The so found non-dominated points might contain duplicates and also might not resemble the complete Pareto front. Therefore, it is necessary to filter for duplicates and perform a refinement of the Pareto front with the remaining non-dominated points. The non-dominated points are thus sorted in nondecreasing order such that $z_1^1 \leq z_1^2 \leq .. \leq z_1^k$. Each consecutive pair of points spans a search rectangle $R(\mathbf{z^i}, \mathbf{z^j})$ with $i \leq j$. These rectangles can now be searched in parallel by the balanced box algorithm (Figure 6.2 (C)). The search rectangles are split in half. First, the bottom half R^B is searched by solving for $\bar{\mathbf{z}}^1 = \operatorname{lex\,min}_{x\in\mathcal{X}}\{z_1(\mathbf{x}), z_2(\mathbf{x}) : z(\mathbf{x}) \in R((z_1^i, \frac{z_2^i+z_2^j}{2}), \mathbf{z}^j)\}$. If a non-dominated point is found, the upper half R^T is further restricted and spans now $R(\mathbf{z}^i, (\bar{z}_1^1 - \epsilon, \frac{z_2^i+z_2^j}{2}))$ in which $\bar{\mathbf{z}}^2 = \operatorname{lex\,min}_{x\in\mathcal{X}}\{z_2(\mathbf{x}), z_1(\mathbf{x}) : z(\mathbf{x}) \in R(\mathbf{z}^i, (\bar{z}_1^1 - \epsilon, \frac{z_2^i+z_2^j}{2}))\}$ is searched for. Each newly found point spans a new independent search rectangle $R(\mathbf{z}^i, \bar{\mathbf{z}}^2)$ and $R(\bar{\mathbf{z}}^1, \mathbf{z}^j)$ with its adjacent point. These rectangles are searched in parallel with the described procedure

(Figure 6.2 (D)). If the search operation yielded the known point $\mathbf{z}^j$ for $R((z_1^i, \frac{z_2^i+z_2^j}{2}), \mathbf{z}^j)$ and $\mathbf{z}^i$ for $R(\mathbf{z}^i, (\bar{z}_1^i - \epsilon, \frac{z_2^i+z_2^j}{2}))$ accordingly, then this proves that the area does not contain further non-dominated points. The rectangle search procedure is carried out until the complete search space has been searched. Note that the rectangle $R(\bar{\mathbf{z}}^2, \bar{\mathbf{z}}^1)$ cannot contain any non-dominated points and, therefore, can be disregarded. The complete description in pseudo code can be found in Algorithm 1.

Approximation

We use an approximation technique introduced by Boland *et al.*[49] which is based on the hypervolume indicator and an adjusted hypervolume indicator introduced by Zitzler *et al.*[188] that allows a quality assessment of an approximated Pareto front.

The measurement indicates the percentage of unexplored search space that could potentially contain undiscovered non-dominated points. Hence, it constitutes a pessimistic bound on the quality of the obtained solution. The search for further Pareto points is stopped, when a certain percentage of unexplored space is reached.

6.2.5 Implementation

JackHMMER[189] was used for MSA construction that formed the basis of the statistical fitness model. To increase residue coverage and sequence diversity, the alignment was created using five search iterations. Sequences with an E-value greater than 10^{-20} and more than 70% gaps, as well as columns with more than 50% gaps where excluded from the subsequent inference of the model. To further reduce sampling bias, the sequences were clustered into bins with 90% sequence similarity, and weighted by the number of cluster members. The parameters of the Potts model and EC scores were inferred using EVfold$^{\text{PLM}}$[158]. Additional structure-based fitness predictions were performed with the FoldX server[190] using the default settings. The resulting bi-objective integer linear program was solved by the newly developed, distributed solver. Statistical analysis was done in R 3.2.1 and in Python using the module SciKit-learn 0.18.

The parallel two-phase balanced box approach was implemented in Python 2.7 using Numpy, Polygon, and the CPLEX Python API. A message passing protocol was implemented using the multiprocessing module of Python to be able to leverage distributed cluster systems. The master process handles the algorithmic procedures, whereas the slave processes implement the lex min-operation. An intermediary broker process, maintaining a job and result queue, was used allowing master and slaves to work independently without any direct communication, which in turn improves the scalability compared to direct message passing between master and slaves, as it allows for asynchronous communication

Algorithm 1: Two-phase balanced box algorithm in pseudo code.

DoneQ.init();
JobQ.init();
List.init();

//init search rectangle;
JobQ.add(lex min$(z_1, z_2, R((-\infty, \infty), (-\infty, \infty)))$;
JobQ.add(lex min$(z_2, z_1, R((-\infty, \infty), (-\infty, \infty)))$;
JobQ.join();

//Phase one - epsilon grid;
$\mathbf{z}^T$,R^T=DoneQ.pop();
$\mathbf{z}^B$,R^B=DoneQ.pop();
List.add($\mathbf{z}^T$);
List.add($\mathbf{z}^B$);
for $i \in [1, N]$ **do**
    $\tau_i = z_1^T + \frac{i \cdot (z_1^B - z_1^T)}{N}$;
    JobQ.add(lex min$(z_1, z_2, R((\tau_i, z_2^T), \mathbf{z}^B))$;
end
JobQ.join();

while *!DoneQ.empty()* **do**
    $\mathbf{z}$,$\mathbf{R}$ = DoneQ.pop();
    List.add($\mathbf{z}$);
end

//Phase two - balanced box refinement;
List.sort();
running = 0;
for $i \in [1, List.length - 1]$ **do**
    $\mathbf{z}^i$ = List.get(i);
    $\mathbf{z}^j$ = List.get($i + 1$);
    JobQ.add(lex min$(z_1, z_2, R((z_1^i, \frac{z_2^i + z_2^j}{2}), \mathbf{z}^j))$;
    running ++;
end

while *running* $\neq 0$ **do**
    islex min$_1, \bar{\mathbf{z}}, R(\mathbf{z}^i, \mathbf{z}^j)$=DoneQ.pop();
    List.add($\bar{\mathbf{z}}$);
    running --;
    if *islex* min$_1$ **then**
        JobQ.add (lex min$(z_2, z_1, R(\mathbf{z}^i, (\bar{z}_1 - \epsilon, \frac{z_2^i + z_2^j}{2})))$;
        running ++;
        if $\bar{\mathbf{z}} \neq \mathbf{z}^j$ **then**
            JobQ.add(lex min$(z_1, z_2, R((\bar{z}_1, \frac{\bar{z}_2 + z_2^j}{2}), \mathbf{z}^j))$;
            running ++;
        end
    else
        if $\bar{\mathbf{z}} \neq \mathbf{z}^i$ **then**
            JobQ.add(lex min$(z_1, z_2, R((z_1^i, \frac{z_2^i + \bar{z}_2}{2}), \bar{z}_2))$;
            running ++;
        end
    end
end

return *List*

between the two. It also enables an efficient management of accumulating tasks, as every solved problem spawns two new subproblems.

6.3 Results

6.3.1 Solver Evaluation

Benchmark Dataset

To evaluate the scalability and runtime behavior of the newly developed two-phase balanced box approach, a benchmark data set was gathered from `http://hdl.handle.net/1959.13/1036183`(accessed March 04, 2016), which consisted of bi-objective, two-dimensional 0-1 Knapsack problems (2DKP) of the following form:

$$\begin{aligned} &\text{(O1)} \quad \max_{\mathbf{x}\in\{0,1\}^n} \mathbf{c}^T\mathbf{x} && (6.11)\\ &\text{(O2)} \quad \max_{\mathbf{x}\in\{0,1\}^n} \mathbf{d}^T\mathbf{x} \\ &\textbf{s.t.} \\ &\text{(C1)} \quad \mathbf{a}^T\mathbf{x} \leq b && (6.12)\\ &\text{(C2)} \quad \mathbf{e}^T\mathbf{x} \leq f \\ &\qquad \text{with} \;\; \mathbf{a},\mathbf{c},\mathbf{d},\mathbf{e} \in \mathbb{R}^n;\; b, f \in \mathbb{R}. \end{aligned}$$

with $n = 375$, 500, 670, up to 750 items, respectively, to select from. The problem instances were designed to consist of a large number of non-dominated points and exhibit similar properties as benchmark datasets used before in the literature[49].

Solver Scalability and Runtime Analysis

All analysis were conducted on an Intel Xeon CPU E7-4850 v2 with 48 processors running at 1.20 GHz, repeating each instance five times. As the single-objective problem instances could be quickly solved, CPLEX was restricted to using one thread to avoid unnecessary overhead. Note however that multithreading should be enabled, if the single-objective reformulation of the problem cannot be solved efficiently.

An instance with 375 items was solved with 1, 2, 4, 8, 16, and 32 workers with a parallel implementation ($\text{BB}_{\text{parallel}}$) and the two-phase implementation ($\text{BB}_{\text{two-phase}}$) of the balanced box algorithm (Figure 6.3).

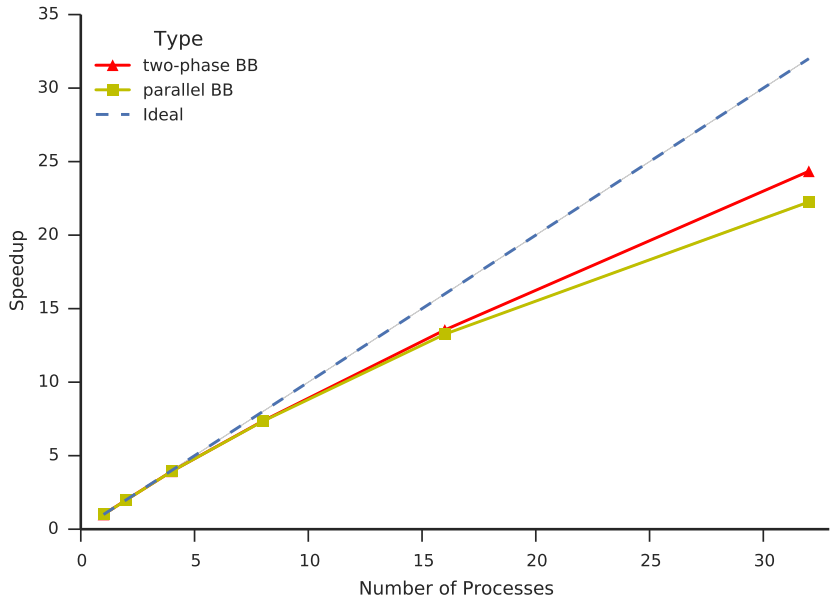


Figure 6.3: Scalability comparison. An instance of the bi-objective, 2D Knapsack problem with 375 items was solved with a parallel implementation of the standard balanced box and the two-phase balanced box algorithm with increasing numbers of cores. The gained speedup is shown compared to the single threaded version of the balanced box algorithm and to the theoretically optimal linear speed (dashed line).

The maximum runtime speedup of 24.42 ± 0.09 and 22.25 ± 0.62 could be achieved for $\text{BB}_{\text{two-phase}}$ and $\text{BB}_{\text{parallel}}$, respectively, using 32 processes. Both versions scaled optimally with a low number of processes (up to four) and started to diverge from the ideal speedup when using more than eight processes. The benefit of the two-phase approach became visible with a larger number of processes (more than 16) and could further decrease the runtime by additional 9% compared to the standard parallel version of the balanced box algorithm. The runtime differences $\text{BB}_{\text{two-phase}}$ and $\text{BB}_{\text{parallel}}$ amounted roughly to the time needed for $\text{BB}_{\text{parallel}}$ to generate the number of identified non-dominated points found by $\text{BB}_{\text{two-phase}}$ in its first phase. Thus, one can expect a much more prominent runtime difference between the two-phase approach and the parallel version of the balanced box algorithm with increasing runtime of the transformed single-objective problems.

The benchmark was extended to include all extracted instances. $\text{BB}_{\text{two-phase}}$ with 32 processes was compared to the single threaded version of the standard balanced box algorithm (Table 6.1). On average, a speedup of 25.59 ± 1.35 could be achieved. With increasing problems sizes, the speedup was more prominent, which might be due to the fact that all processes could be used over a longer period of time in these large instances than in the smaller instances.

Instance	#Items	Runtime (p=1) [s]	Runtime (p=32) [s]	Speedup
A2	375	2526.11	103.45	24.42
A3	375	4749.48	190.58	24.92
A4	375	2542.89	111.60	22.79
A5	375	3655.78	151.17	24.18
B7	500	11183.04	436.82	25.60
B8	500	8741.45	334.73	26.12
B9	500	5868.75	231.24	25.38
B10	500	6783.91	271.89	24.95
C11	670	14866.88	567.41	26.20
C12	670	15085.83	579.13	26.05
C13	670	17114.68	633.67	27.01
C14	670	11870.85	454.90	26.10
C15	670	10869.00	417.11	26.06
D20	750	19033.68	668.93	28.45

Table 6.1: Runtime and speedup analysis of the two-phase balanced box algorithm.

6.3.2 Application: De-immunization of Factor VIII

Factor VIII and Hemophilia A

To illustrate the model's capability, we used factor VIII as use case. Factor VIII is essential for blood clotting. Its gene is located on the long arm of the X chromosome (Xq28), being approximately 186 kbp long and consisting of 26 exons and introns[191]. The gene product comprises 2,332 AA and is structured into six domains, A1-A2-B-A3-C1-C2, of which A1-A2 and parts of the B domain form the heavy chain and A3-C1-C2 the light chain[192]. The protein is produced in sinusoidal cells of the liver as well in endothelial cells of the whole body[193]. Bound to the von Willibrand factor (vWF), it circulates in the blood. Upon injury of blood vessels, it separates from vWF and is activated via proteolysis of both the heavy and light chain[193]. Factor VIII recruits and forms a complex with factor IX, another coagulation factor. The complex formation leads to the activation of factor IX and factor X, which initialize a positive feedback cascade that ultimately results in the formation of stable fibrin clots[194].

Gene defects in factor VIII result in the development of hemophilia A, of which 67% are nonsynonymous point mutations, 25% are caused by small deletions and insertions, and 6% are large deletions[195]. In 40%-50% of patients suffering from severe hemophilia A, it is caused by an inversion gene defect in intron 22, which leads to a complete disruption of the protein. This gene defect can be traced back to errors in DNA replication during spermatogenesis of the male parent[196]. These gene defects lead to hindered clot formation and subsequently to haemorrhagic diathesis, which manifests in prolonged bleeding

episodes, and spontaneous bleeding into soft tissues, joints, and muscles. Repeated haemorrhages were reported to cause chronic arthropathy, joint deformation, and loss of joint movement [197,198].

The only on-demand or prophylactic treatment currently available is the administration of functioning factor VIII to establish homeostasis. The treated dosage, frequency, and number of infusions are strongly dependent on the severity of the illness. Prophylactic treatment to prevent bleeding and joint damage has become a standard procedure, starts at very young age (≤ 2 years), and is nowadays even tailored to the personal needs of the patient often times using pharmacokinetic models and computer simulations [199].

The treatment, especially with prophylactic therapy, can be severely hampered by the formation of anti-drug antibodies (ADAs). In 10-15% of patients with mild, and in 30% of patients with severe hemophilia A develop ADA [200]; hence patients with the highest need for therapy are those least likely to benefit. This correlation between severity of the disease and lack of efficacy follows from the fact, that the body is more likely to recognize the therapeutic factor VIII as foreign the more severe the natural mutation. In the extreme, mutations that cause a total loss of factor VIII production are most strongly related to ADA development [201,202]. The process of ADA development has been shown to be T-cell dependent [203] directed by the presentation of factor VIII-derived peptides on HLA molecules of the patient. Driven by these serious side-effects, we focused our de-immunization efforts on the C2-domain of factor VIII, which has been shown to be highly immunogenic and involved in ADA development [204,205].

De-immunization of Factor VIII

To identify immunogenic clusters the factor VIII C2-domain sequence (UniProt: FA8_HUMAN, residues 2,188-2,345) was screened with TEPITOPEpan with the three most prevalent HLA alleles in the European population DRB1*03:01, DRB1*07:01, and DRB1*15:01 with a binding threshold of 5%. The screening identified a large region starting at residue 2,312 and ending at residue 2,340 (Figure 6.4 (A)). In total, sixteen epitopes, of which nine epitopes were located in the identified region, were predicted to bind to at least one of the three HLA alleles. Also, three of the five most interacting residues resided in the region. The region was in general significantly enriched with predicted strongly coupled residues compared to other regions of the same size (sign test, s = 124, n = 130, p-value $<$ 2.2e-16, CI_{95} = [0.91, 1.00]). It is structurally characterized by a long beta-sheet as well as a loop region (Figure 6.4 (C)). Also, the region starting at residue 2,332 to residue 2,334 has been described as an important component of the membrane binding motif [206].

De-immunization efforts were thus focused on the identified immunogenic region of factor VIII. The described model was used to calculate the Pareto fronts of mutation

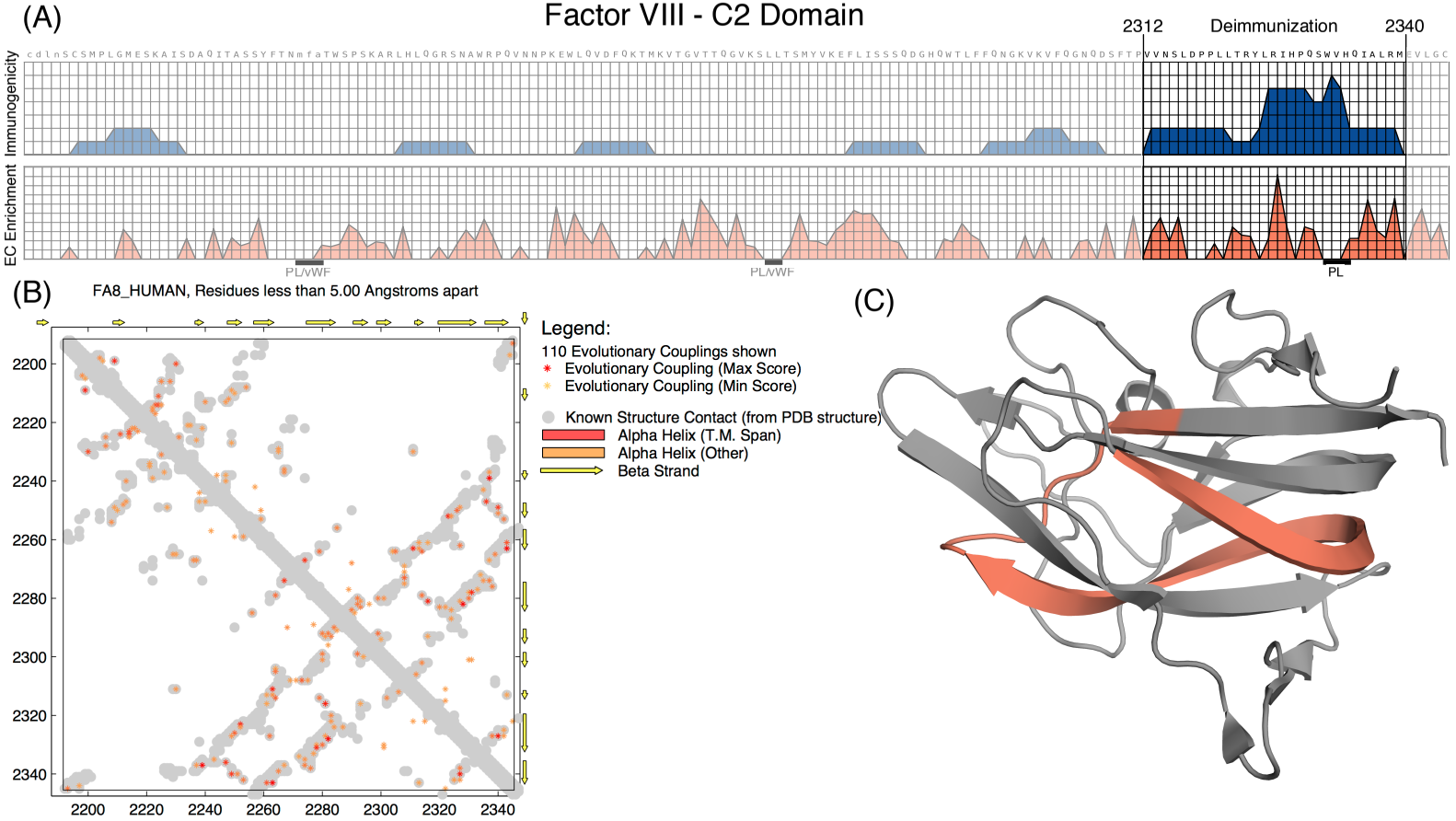


Figure 6.4: (A) Immunogenicity screening for three DRB1 alleles with TEPITOPEpan. The blue density plot depicts the cumulative immunogenicity scores per position. The orange density plot depicts the top 90 contact enriched residues scores. The highlighted region starting at residue 2,312 and ending at 2,340 marks the most immunogenic region and the target of the deimmunization. (B) Contact map of FA8_HUMAN. The grey fragments indicate experimentally determined structure contacts, whereas the red and orang dot represent predicted contacts. (C) Tertiary structure of factor VIIIs C2 domain (PDB: 3hny). The orange-colored segment marks the strongly immunogenic cluster starting at residue 2,312 and ending at 2,340. Note that this structure was used for visualization only and is not required for the calculation.

ID	Mutation	Epitopes	ΔImmunogenicity	ΔHamiltonian
wt		16		
0	V2333E	11	-0.38	1.14
1	L2321F	16	2.25	0.83
2	Q2335H	16	4.91	0.77
3	Y2324L,V2333E	9	-2.16	6.47
4	Y2324H,V2333E	10	-1.84	5.96
5	R2326K,V2333E	10	-1.84	4.31
6	L2321T,V2333E	10	-1.59	3.68
7	L2321Y,V2333E	11	-1.01	3.48
8	L2321F,V2333E	12	-0.99	1.97
9	V2333E,Q2335H	12	0.43	1.93
10	L2321F,Q2335H	17	4.3	1.47
11	V2313M,Y2324L,V2333E	8	-2.52	7.99
12	L2321T,I2327L,V2333E	8	-2.39	6.47
13	L2321F,R2326K,V2333E	10	-2.21	5.32
14	V2313M,L2321T,V2333E	9	-1.95	5.16
15	L2321F,I2313V,V2333E	10	-1.92	4.92
16	L2321F,I2313L,V2333E	10	-1.53	4.61
17	V2313T,L2321F,V2333E	10	-1.36	4.58
18	V2313M,L2321F,V2333E	11	-1.34	3.52
19	L2321F,Y2324F,V2333E	12	-1.05	3.26
20	L2321F,V2333E,Q2335H	13	-0.19	2.62
21	L2321F,Y2324F,Q2335H	17	4.24	2.55

Table 6.2: De-immunization results for mutation loads of k=1,2,3.

loads between 1 and 3 simultaneous point mutations. Only residues with a site-specific occurrence of more than 1% were considered as potential substitutions. TEPITOPEpan with a binding threshold of 5% was used as internal prediction method for immunogenicity prediction. Sequence alterations were only allowed to appear in the described region, but were still selected based on the global alternation of the fitness landscape (i.e., all network constraints of the fitness model were considered).

The de-immunization yielded sequence alterations shown in Table 6.2. The trade-off between immunogenicity and the fitness objectives was strongly visible. The less immunogenic the designs became, the stronger effected was the fitness of the construct. However, the fitness of the designs remained in close proximity of the wild type fitness with an average distance of $0.81 \pm 0.43\%$. In general, no design yielded a stabilizing effect suggesting that the wild type is close to an energetically optimal conformation within the defined design space.

The maximal reduction of immunogenicity could be achieved with design 11 (V2313M, Y2324L, V2333E), yielding an immunogenicity reduction of 44.99% of the whole domain deleting 8 out of 9 epitopes within the selected region by simultaneously decreasing the protein's fitness by only 1.71%. The next best triple mutant (L2321T, I2327L, V2333E) also achieved a deletion of eight epitopes by an immunogenicity reduction of 42% and destabilization by 1.28%. Generally, all fitness scores resided in 95% percentile or higher except of design 11 (V2313M, Y2324L, V2333E), which was located in the 90% percentile.

Experimental Immunogenicity Evaluation

In order to experimentally verify the predicted mutants, 15- to 16-mer peptides were designed to span the predicted mutation sites and most of the predicted epitopes influenced by the introduced mutation. As control their wild type counterparts were also tested. The so designed peptides covered all designs with one mutation as well as a selection of double and triple mutant designs (Appendix Table F.2).

The peptides were tested with a commercial REVEAL HLA-Peptide binding assay of ProImmune (`www.proimmune.com`) for the same HLA alleles used in the *in silico* de-immunization. The peptides were synthesized with the PEP-screen custom library method. The used experimental method compares the affinity of the custom peptides to the affinity of a known and highly affine reference peptide. The affinity of the peptide in question is then reported as percentage of the signal generated by the control peptide. The affinity is measured twice at time point H0, at the beginning of the incubation and again after 24 hours (H24) (Appendix Table F.3).

The allele-specific scores were summarized by linearly combining the measured affinity scores of each HLA allele (Figure 6.5 (C)). As predicted, most of the introduced mutations

reduced the overall immunogenicity of the peptides. The highest reduction of 86.04% was achieved by M-8 at H0. The strongest increase in immunogenicity of 87.01% at H0 was observed for M-4 confirming the predictions (predicted increase of 86.37%). Overall, the measured and predicted immunogenicity of the tested peptides correlated strongly at H0 with r = 0.76 (CI_{95} = [0.47, 0.90], t = 4.85, df = 17, p-value = 1.5e-4, Figure 6.5 (A)).

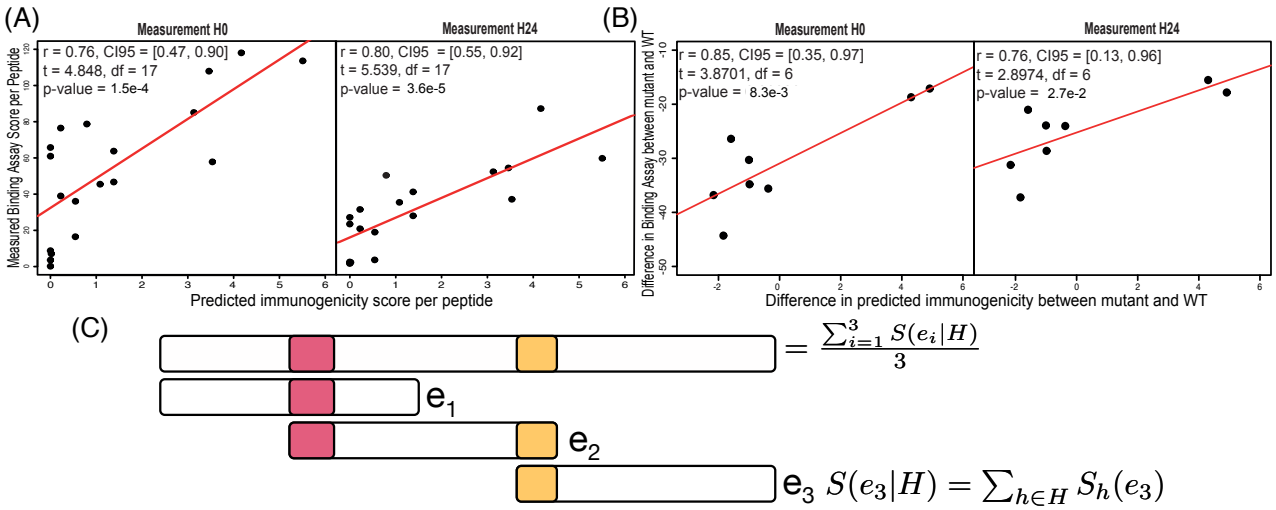


Figure 6.5: (A) Correlations of measured and predicted immunogenicity of each designed peptide of factor VIII at time point H0 and after 24h H24. (B) Correlations of the approximately measured and predicted change in immunogenicity for the complete region of interest at both time points. (C) Illustration of how the measured immunogenicity scores S of each epitope e_i and HLA allele $h \in H$ were combined. Colored bars represent introduced mutations.

To compare the predicted and measured gain or loss in immunogenicity caused by the introduced mutations of the complete designs, overlapping peptides were used to reconstruct the targeted region. The measured scores were again linearly combined and normalized against the number of overlapping peptides used for reconstruction (Figure 6.5 (C)). The difference between wild type and mutant of the so calculated scores resemble the approximated gain or loss in immunogenicity of the complete region. These tendencies were compared with the differences between predicted wild type and mutant immunogenicity for both time points (Figure 6.5 (B)). At time point H0, a strong linear correlation of r = 0.85 could be observed (CI_{95} = [0.35, 0.97], t = 3.87, df = 6, p-value = 8.3e-3). Similar results were obtained for the data set collected at time point H24.

Evaluation of Functional Predictions

In a next step, we validated whether the evolutionary Hamiltonian can be estimated accurately for factor VIII. As the alignment used for inference contained 4,800 sequences,

we attained a sequence coverage of $30 \times n$ ($n = 157$ AA, the length of the C2 domain), which should suffice for high quality inference.

To validate the quality of the inferred model, we used the spatial accuracy of the total epistatic constraints between residue pairs as an approximation of the model's validity assuming that correctly predicted distance constraints of residue pairs are a direct measure of the model's quality. We therefore compared predicted distances of the top 90 residues to a known crystal structure of the factor VIII's C2 domain[206] (pdb: 3hny, Figure 6.4 (A)). Seventy-nine out of 90 ECs of the two residue pairs were spatially close (below 5 Å) in the used crystal structure, which corresponds to a model precision of 83%.

To further show the applicability of the Hamiltonian to predict structural effects of mutations, we used the Hamiltonian model in a multinomial and logistic regression to predict hemophilia A severity based on patient data collected from the factor VIII variant database (`http://www.factorviii-db.org`). Since the severity of hemophilia A is directly correlated with instability and malfunctioning of factor VIII, the prediction of disease severity based on Hamiltonian changes can be seen as a proxy for functional and structural effect prediction. A multinomial linear regression model was fit to single point mutation data with known severity status that resided in the C2 domain of factor VIII (Appendix Table **??**). The change of Hamiltonian was used as independent variable, while the dependent variable was categorized into three severity classes (severe, moderate, and mild) based on a one-stage factor VIII:C assay. The data were randomly divided into training and test set (70:30%-split) in a stratified manner. This process was repeated two hundred times and the performance averaged over the runs. The multinomial regression model achieved moderate prediction performance with a F1-micro score of 0.65 ± 0.09, a F1-macro score of 0.47 ± 0.07, and a log-loss of 0.95 ± 0.12. We combined the severe and moderate class, and performed a logistic regression based on the same training and testing procedure. The logistic regression model achieved good performance with a weighted AUC of 0.72 ± 0.11, a weighted F1-score of 0.73 ± 0.11, and a log-loss of 0.63 ± 0.06. By disregarding the moderate class, the prediction performance could be further increased to weighted AUC of 0.75 ± 0.11, weighted F1-score of 0.74 ± 0.12, and log-loss of 0.59 ± 0.06.

We also compared the predicted changes of the Pareto front designs to a commonly used structure-based approach namely FoldX (Figure 6.6). This method relies on structure, rather than sequence, and predicts mutational effects on proteins based on a force field, thus providing orthogonal information to the employed sequence-based approach. A significant correlation of r = 0.44 (CI95 = [0.02, 0.73], t = 2.17, df = 20, p-value = 4.2e-2; Figure 6.6 (A)) could be observed. The two most deviating mutations were design 11 (V2313M, Y2324L, V2333E) and design 3 (Y2324L, V2333E), both of which introduced a mutation at a membrane binding position[206]. This is why FoldX was under-predicting the deleteriousness of these two designs, as a force field-based approach cannot capture such functional relations.

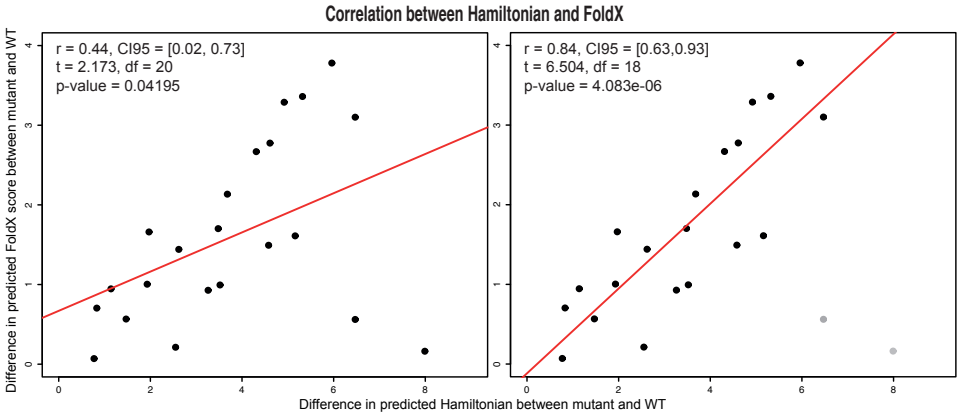


Figure 6.6: Correlation between predicted mutational influence using the maximum entropy model and FoldX before (left) and after (right) outlier correction.

In contrast, the Hamiltonian-based model was able to capture these relationships. After removing these two outliers, a strong correlation of r = 0.84 (CI95 = [0.63,0.93]) could be observed between the Hamiltonian-based predicted influences and the FoldX predictions (t = 6.50, df = 18, p-value = 4.1e-6; Figure 6.6 (B)).

Taking together the collected evidence, we are confident that the evolutionary Hamiltonian captures and predicts the protein fitness of mutational changes with good accuracy.

6.4 Discussion

Immunogenicity and the formation of anti-drug antibodies (ADAs) is a major problem of all classes of biotherapeutics. Large, time-consuming experimental screening efforts have to be conducted to find a few suitable sequence modifications to reduce the immunogenicity of the biotherapeutic without major impediment of its structure or function. Computational methods can help to guide these screening experiments, thus allowing to explore a larger design space while reducing time and cost expense. Here, we introduced a computational method that finds sequence alterations to reduce the protein's immunogenicity while maintaining its stability requiring only sequence information of the target protein. The method uses a new immunogenicity objective that is, on the contrary to previous approaches, quantitative and integrates HLA allele distribution information of a target population. Since the HLA distribution differs tremendously between populations, this will influence the immunogenicity of a protein and hence T-cell epitopes that should be prioritized during the de-immunization process.

To solve the resulting bi-objective combinatorial problem, we developed a new parallel solving strategy that efficiently exploits modern distributed computing systems. Our benchmark experiments showed that the implementation is highly effective yielding average

speedups of 25-fold. Also the results suggest that the two-phase approach taken to overcome some problems of the early stages of the balanced box algorithm could be highly beneficial when using a large amount of processes and if the transformed single-objective problems do need a significant amount of time to be solved. The efficacy of the solver can be further increased by using the multi-threading capabilities of several single-objective solvers, such as CPLEX, to even further exploit the structures of modern distributed cluster, grid, and cloud systems. However criterion-space search algorithm and particular the balanced box algorithm elicit a weakness. Due to the fact that these algorithms in general manipulate floating point values multiple times, and in the case of the balanced box method even compare for equality, inaccuracies can arise which can lead to numerical instabilities, if the floating point accuracy parameters are not carefully chosen.

While previous approaches for Pareto-optimized de-immunization[21,152] relied on the existence of solved tertiary structures and force-field based approaches for mutational free energy prediction, such as FoldX, we demonstrated that at least a similar performance can be achieved using the evolutionary record of the target protein alone given that there is a sufficient amount of diverse sequence information available. The high precision of the predicted evolutionary couplings compared to a solved 3D structure showed that the sequence-based global co-evolution model is sufficiently accurate for fitness prediction. The strong correlation between fitness changes predicted by the hamiltonian and FoldX stability predictions, as well as the good performance in predicting hemophilia A severity further supports this assumption.

In a proof-of-principle study we applied our method to de-immunize the C2 domain of factor VIII. Factor VIII is used as substitution therapy in hemophilia A patients, albeit adverse immune reactions and ADA-formation are frequently observed exacerbating the therapy tremendously. The C2 domain has been shown to be highly immunogenic and involved in ADA formation[150,205]. Thus, by identifying and removing epitope clusters via de-immunization within the domain could decrease ADA formation and thus side effects of the biotherapeutic. The fact that the identified epitope cluster coincided with a highly evolutionary connected as well as functional important region underlines the need for intelligent methods that are capable of incorporating structural and functional integrity prediction in the de-immunization process. The *in silico* de-immunization step using the proposed method demonstrates the power of such approaches; the immunogenicity of the complete domain could be reduced by 44.99% by only focusing on the most immunogenic region without disrupting the fitness landscape extensively. Moreover, the observed highly significant correlations between measured and predicted immunogenicity both on individual peptide and (reconstructed) region level affirmed that the underlying assumptions made by the model are sufficient enough to predict the influence of mutation in terms of immunogenicity.

The next step in an experimental validation of the factor VIII variants described here could involve an *in vivo* mouse model as the one used by Moise *et al.*[150]. Another alternative would be to use PBMCs from hemophilia A patients with known inhibitor status to study T-cell proliferation. Furthermore, it would also be very intriguing to further investigate the effects on a personal level, studying mutations within the factor VIII gene, inhibitor status, and HLA type of the patient.

While the utilization of the protein's evolutionary record can be of use in cases when the crystallization of the target proteins poses a problem, its dependence on a strong evolutionary background can be problematic for target molecules that are chimeras or engineered. In this case, it is currently not possible to obtain a sufficiently large and diverse record for model inference. Thus, a combination of both sequence-based and structure-based stability approximation methods could be combined to overcome the problems of both approaches. Also, all proposed models can only indirectly predict the functional impact of newly introduced mutations on interacting protein partners. Structural information of interaction partners could be used to identify interacting residues and subsequently exclude these from the pool of potential mutation sites. However, such data is scarce. Recently, the used statistical model was generalized to predict interacting residues of protein-protein interactions based on sequence information alone[157]. Such predictions could also be directly incorporated into the fitness objective or used to exclude interacting positions as possible mutation sites, if crystal structures of the interacting partners are not available.

In summary, we proposed a novel de-immunization model that integrates quantitative immunogenicity optimization with sequence-based fitness optimization. We then demonstrated the use of this model and the validity of our underlying assumptions by comparing predictions for human factor VIII de-immunization to experimentally determined immunogenicity scores and well established structural-based stability prediction methods. Hence, this approach will allow bioengineers to reliably explore the design space of the target protein to select promising candidates for experimental evaluation.

Chapter 7

Translational Immunoinformatics

7.1 Introduction

Computational immunology has significantly matured over the last decade and its applications are now widely used in biomedical research, especially in the field of basic cancer and applied immunotherapy research[11,88]. These applications often require complex pipelines, including pre- and post-processing, and use many different tools. The lack of standardized data formats and interfaces in the immunoinformatics community makes the development of such pipelines and the interoperability of various prediction tools difficult. Also, most state-of-the-art immunoinformatics software has been developed for Unix-based operation systems only and involves complex installation procedures. It is thus often challenging for inexperienced researchers to apply these tools to their biomedical question.

Only a few attempts have been made to overcome these issues. The framework for epitope detection (FRED)[207] tries to unify interfaces to several prediction methods by building a Python-based framework around them. It allows for rapid development of complex immunoinformatics pipelines and easy interchangeability of different prediction methods. The authors of Epitopemap[208] use a similar idea and developed a Python interface for the IEDB supported epitope prediction methods together with a web-based platform for epitope prediction and visualization.

Building on the idea of an unifying framework, we modernized FRED by completely re-implementing and significantly extending its functionality, covering pre- and post-processing, epitope binding, antigen processing prediction, HLA inference, and vaccine design (Section 7.2).

While FRED partly solves the issue of interoperability and rapid development, the installation and usage remains problematic for the average biomedical researcher. Web services, like the ones offered by the Center Biological Sequences (CBS) for their NetMHC prediction family (`http://www.cbs.dtu.dk/biotools/`), or the Immune Epitope Database

(IEDB)[209] Analysis toolbox (`http://www.iedb.org/`) could solve the requirement of installing multiple software suites. However, these web services only offer little guidance to the inexperienced user. EpiToolKit[210] represents a notable exception. It carefully guides the user step-by-step through the configuration of the tool with detailed explanations. But none of these web services allow for a smooth interaction between different tools to build analysis workflows. That is why we re-implemented and extended EpiToolKit to enable such advanced functionality without losing its philosophy (Section 7.3).

Factors such as data volume, speed, or legal restrictions (e.g., data privacy), often prevent the use of web-based solutions, especially in biomedical research. These web services can also be unreliable in terms of reachability and they can be slow depending on the infrastructure of the web service. To meet the demands and reliability of biomedical research, we developed *ImmunoNodes* (Section 7.4), an immunoinformatics toolbox that is tightly integrated into the Konstanz Information Miner Analytics Platform (KNIME)[23,24], an application for visual workflow development. It allows users to build complex workflows with an easy to use and intuitive interface with a few clicks on any desktop computer. Together with the KNIME Server and Grid Engine[24], users can utilize their local distributed computing system to scale-up their application if needed.

7.2 FRED 2 - An Immunoinformatics Framework for Python

Parts of this chapter were published in:

Schubert, B., Walzer, M., Brachvogel, H. P.,* et al. *(2016).*
FRED 2: an immunoinformatics framework for Python.
Bioinformatics, 32(13) 2044-2046.

FRED 2 (FRamework for Epitope Detection) is an open-source, Python-based framework for computational immunology. It is the completely re-implemented successor of FRED[207] and provides a unified interface to many immunoinformatics related prediction tools. We implemented routines covering data pre-processing, HLA typing, epitope and antigen processing prediction, epitope selection, as well as epitope assembly. FRED 2 is flexibly designed to allow easy extension by providing well-defined interfaces. Building on top of popular modules such as BioPython (`http://biopython.org`) and Pandas (`http://pandas.pydata.org`), FRED 2 allows for rapid prototyping of complex and innovative immunoinformatics applications.

7.2.1 Implementation

FRED 2 is divided into four major packages: *Core*, *IO*, *Vaccine Design*, and *Prediction* (Figure 7.1). In *Core*, classes are found that represent the most important biological entities *Transcript*, *Protein*, *Peptide*, and HLA *Allele*, as well as *Variant*. It also provides pre-processing functions to integrate *Variants* into *Transcripts* and generator functions to cast one entity into the other if appropriate.

IO provides functionalities to read standard biological file formats such as FASTA, or ANNOVAR[211], as well as Variant Effect Predictor[212] generated VCF files and provides interfaces to major databases such as UniProt[213], RefSeq[214] and Ensemble[215] via Biomart[216]. All database adapters have a unified interface *ADBAdapter* enforced through Python's *AbstractBaseClass*[217].

Prediction methods are split into four packages *EpitopePrediction*, *TAPPrediction*, *CleavagePrediction*, and *HLATyping*, each providing factory classes as single entry points for the supported prediction methods (detailed overview of all supported prediction methods can be found in Appendix Table F.5). These factory classes serve as registration of all prediction methods of a single type (e.g., epitope prediction). Newly implemented prediction methods are automatically registered in the corresponding factory class via metaprogramming by correctly inheriting from the suitable metaclass interface. The prediction methods return *AResult* objects, which are functionally extended *Panda.DataFrames*.

FRED 2 also offers functionality for rational vaccine design. It implements *OptiTope*, the mathematical framework for epitope selection proposed by Toussaint *et al.*[105], as well as the epitope assembly approach suggested by Toussaint *et al.*[101] and the one discussed in Chapter 5. We additionally extended the TSP method by Toussaint *et al.* to a bi-objective problem, called *ParetoEpitopeAssembly*, in which the first objective optimizes the cleavage likelihood as before, while the second objective minimizes the neo-epitope counts that are pre-computed for any epitope pair with a supported epitope prediction method. We solve the bi-objective model with the standard ϵ-constraint method (Section 3.2.2).

FRED 2 is open-source (`http://fred-2.github.io/`) and released under a three-clause BSD license. It is designed to be open and easily extendable by providing self-explanatory interfaces using abstract base classes so that implementation of new functionality by a wider community can be easily accomplished.

7.2.2 Application

Using FRED 2, complex pipelines can easily and quickly be implemented. To demonstrate FRED 2's capabilities, we re-implemented the minor histocompatibility antigen (miHA) identification pipeline described by Fehldhahn *et al.*[218]. Minor histocompatibility antigens play a crucial role in transplantation settings especially in hematopoietic stem cell trans-

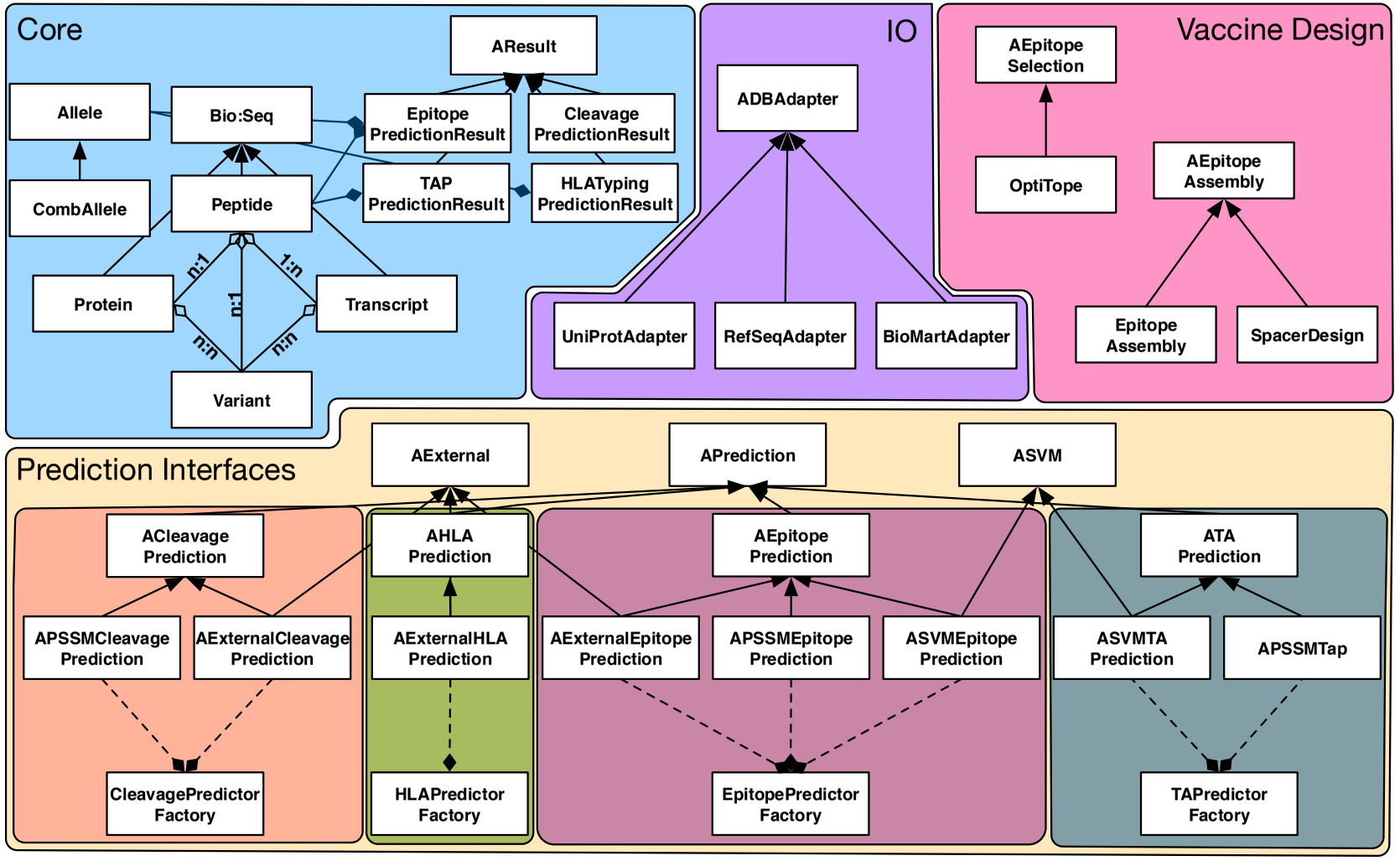


Figure 7.1: Simplified UML diagram of FRED 2. FRED 2 can be divided into four major packages that provide classes and functions for recurring tasks in immunoinformatics. All packages provide pre-defined interfaces in the form of abstract base classes (indicated with a leading *A*) to guide extensions.

plantation (alloHCL) as a treatment for certain hematologic malignancies[219]. They are often the cause for graft rejections in transplantation or graft-vs-host diseases in alloHCL, but can also be used to support traditional cancer therapy by exploiting their beneficial graft-vs-leukemia or graft-vs-tumor effects induced by donor T-cells[220].

Listing 7.1: Individualized miHA identification pipeline using FRED 2.

```
from Fred2.Core import Allele, generate_peptides_from_variants
from Fred2.EpitopePreidction import EpitopePredictorFactory
from Fred2.IO import read_annovar_exonic, read_lines, MartsAdapter, EIdentifierTypes
from operator import ge

#initialize Biomart Adapter
marts = MartsAdapter()

#read matched HLA alleles
hlas = read_lines("matched_hlas.tsv", in_type=Allele)

#read donor and patient variants
donor_vars = read_annovar_exonic("donor_variants.vcf")
patient_vars = read_annovar_exonic("patient_variants.vcf")

#generate peptides and filter for potential miHA epitopes
donor_pep = generate_peptides_from_variants(donor_vars, 9, marts, EIdentifierTypes.ENSAMBLE)
patient_pep = generate_peptides_from_variants(patient_vars, 9, marts, EIdentifierTypes.ENSAMBLE)

candidate_pep = set(patient_pep)-set(donor_pep)

#init epitope prediction method, predict binding affinity for candidate_pep,
#and filter for binders (threshold ≤ 500nM)
netMHCpan = EpitopePredictorFactor("netmhcpan")
filtered_binding = netMHCpan.predict(candidate_pep, alleles=hlas).filter_result((netMHCpan.name,ge,0.425))

#write results to file
filtered_binding.to_csv("candidate_miHA.tsv")
```

The aim of this pipeline is to identify potential miHA epitopes of an HLA-matched donor-patient pair for T-cell priming to increase the graft-vs-leukemia effect after bone marrow transplants. Given the variants of the patient and donor for genes relevant in the hematopoiesis as well as the matched HLA alleles, one can generate the peptides of both donor and patient of the relevant genes and filter for peptides that are unique to the patient and bind to at least one of the matched HLA alleles. Such a pipeline can be easily implemented in a few lines of code using FRED 2 (Listing 7.1).

7.3 EpiToolKit - A Web-based Workbench for Vaccine Design

Parts of this chapter were published in:

Schubert, B., Brachvogel, H., Jürges, C., and Kohlbacher, O. (2015).*
EpiToolKit - A Web-based Workbench for Vaccine Design.
*Bioinformatics, **31**(13), 2211-2213.*

EpiToolKit (ETK) 2 is a web-based platform for computational vaccine design and other immunoinformatics related applications. It supports every design step from HLA genotyping of individuals, epitope discovery, epitope selection, to epitope assembly and can be used for personalized or population optimized vaccine development. ETK 2 is based on a customized version of the open-source platform Galaxy, which allows for a flexible combination of tools as workflows, a reliable recording and sharing of results, and the interaction with high-performance computing resources. In close resemblance to the old implementation, ETK 2 also offers "all-in-one" versions of the tools that guide the inexperienced user through each step of the configuration.

7.3.1 Implementation

ETK 2 was designed to ease the use for inexperienced users while still retaining great flexibility in combining the different tools Galaxy offers. To accomplish this, ETK 2 is divided into two sections.

Under *Single Tools*, Galaxy's interface has been heavily customized to simplify the configuration process of the different tools. The input pages are separated into several configuration steps. The user is guided with individual help texts in each configuration step, similar to its predecessor ETK (Figure 7.2). Each tool is also accompanied by an extended help page in ETK 2's wiki. Galaxy's standard interface was extended with custom *Back* and *Next* buttons, that allow the user to navigate between the different configuration steps without losing previously entered configurations. Once a tool has been fully configured and the task is submitted, its progress can be monitored on the *History* panel.

Under *Workflow*, these configuration steps are available as independent nodes, allowing the development of complex workflows using Galaxy's excellent graphical workflow editor.

EpiToolKit's Tools

ETK 2 offers currently six immunoinformatics-related tools all implemented using FRED 2 (Section 7.2):

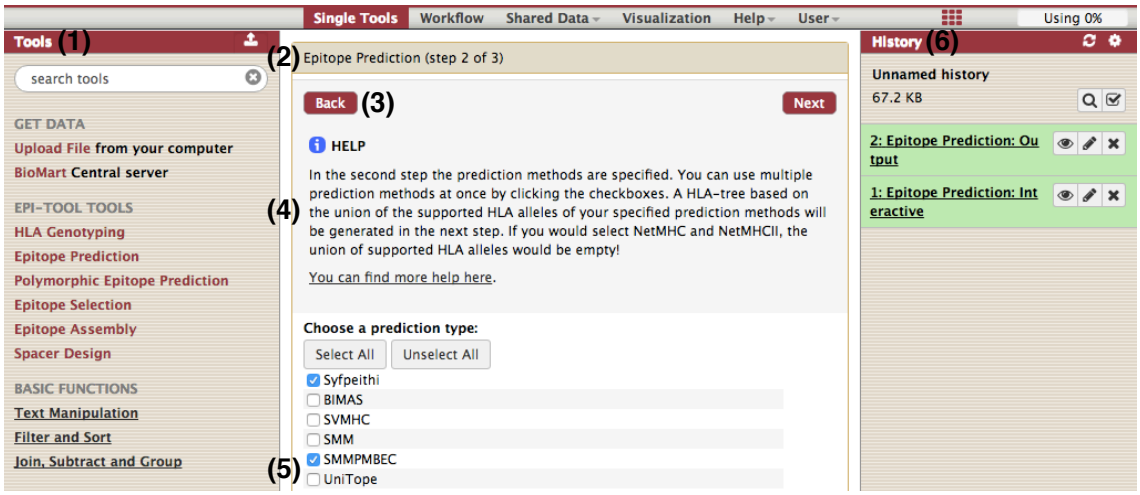


Figure 7.2: Screen shot of ETK 2's web-interface. (1) The left panel lists all available tools. (2) The middle panel visualizes the tool configuration and results pages. In *Single Tools* view, the tool configuration is separated into several steps which can be navigated via custom build buttons (3). Each step offers a help text (4) to guide the users' configurations (5). All submitted tasks are then displayed on the right *History* panel (6).

Epitope Prediction: summarizes several HLA I and II epitope prediction tools into one user interface. Different sequence input options provide access to protein databases like NCBI RefSeq[214] and UniProt[213] in addition to manually entered protein or peptide sequences.

Polymorphic Epitope Prediction: extends *Epitope Prediction* by incorporating variant information. From these variants, neo-antigens are constructed which enables the discovery of neo-epitopes that are influenced by the used variant information. Variant information can either be retrieved from dbSNP[221] or from ANNOVAR[211] generated VCF files. *Polymorphic Epitope Prediction* is based on SNEP[222] and was extended to handle indels and frame shift mutations beside single nucleotide polymorphisms.

HLA Genotyping: provides an interface to OptiType, a novel approach for HLA genotyping based on NGS data (Chapter 4).

Epitope Selection: is an interface to OptiTope[105,223], a highly flexible mathematical framework for epitope selection. It selects a set of k epitopes that maximizes the overall immunogenicity and thus the probability of inducing a long lasting immunity under certain user-defined constraints.

Epitope Assembly: is an implementation of Toussaint *et al.*'s traveling salesman formulation[101] of the epitope assembly problem discussed in Chapter 5.

Spacer Design: provides an interface to the epitope assembly and spacer design approach discussed in Chapter 5.

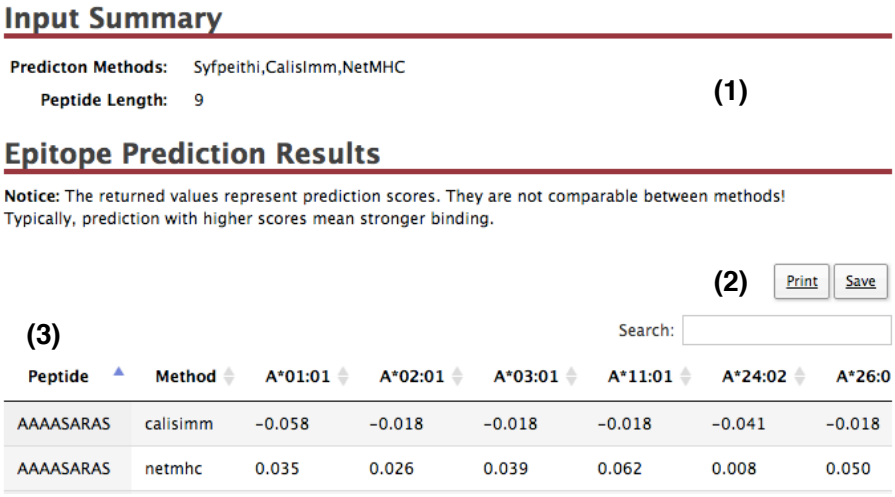


Figure 7.3: Screen shot of ETK 2's epitope prediction result page. The result page offers a configuration summary (1), search and export functionality (2), and an interactive, sortable results table (3).

In the *Workflow* view, additional tools become available that are usually included as a configuration step of the tools available under the *Single Tools* view:

Epitope Conservation: consumes a multiple sequence alignment of proteins and calculates the consensus sequence. Based on this, it generates a list of k-mers and their conservation scores, which is defined as the product of column-wise conservation of the multiple sequence alignment [105]. If an epitope could have originated from multiple sites, the maximum epitope conservation is taken. The epitope conservation output can be used in *Epitope Selection* to filter weakly conserved epitopes.

Epitope Filter: consumes the internal output of an epitope prediction task and filters the predictions based on a user defined threshold.

Allele Selection: offers predefined lists of HLA alleles prevalent in different populations and geographic regions. It also allows to enter or select a user defined HLA list.

Allele Frequencies: consumes a HLA allele list and assigns allele frequencies based on a selected population or geographic region. It also allows to assign custom frequencies to the input alleles, or allocates a uniform probability per HLA locus.

All tools generate two outputs: an interactive presentation of the results in HTML and an internal representation. The internal representation is a simple tab-delimited format

that can be used as input to other tools provided by ETK 2 or Galaxy enabling the user to build complex analysis workflows with Galaxy's visual workflow editor. The HTML display is making use of AJAX and jQuery libraries to allow high responsiveness and interactivity. The HTML result pages offer a static configuration summary, as well as an interactive table that can be sorted, searched, and exported to CSV or Excel (Figure 7.3).

7.3.2 Application

To demonstrate ETK 2's capabilities, we developed a workflow for designing population-optimized vaccines for seasonal influenza (Figure 7.4).

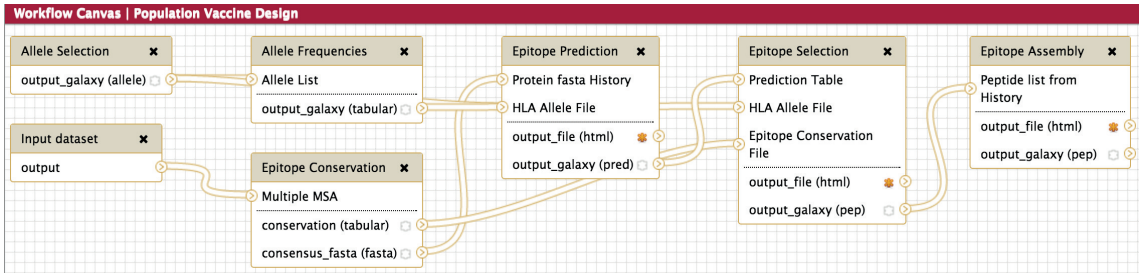


Figure 7.4: Example workflow for population-based vaccine design. *Allele Selection* allows for specifying the target population represented by their HLA alleles. *Allele Frequencies* then assigns frequencies to the chosen HLA alleles based on pre-assembled data or manually assigned frequencies. *Epitope Conservation* takes a file containing multiple MSA of antigens and constructs consensus sequences for each of them and calculates conservation scores for each k-mer peptide generated from the consensus sequences. *Epitope Prediction* performs the epitope prediction for the specify HLA alleles and the consensus sequences. *Epitope Selection* consumes the prediction results and selects a pre-defined number of epitopes under constraints for the specified target population and antigens. *Epitope Assembly* arranges the selected epitopes such that their recovery probability after proteasomal cleavage is maximal.

Based on the yearly WHO recommendations, a dataset consisting of H1N1 and H3N2 strains was extracted from the Influenza Research database[224]. Using NetMHC[225] and default configurations for the epitope selection step, 10 epitopes were selected (Appendix Figure E.2). The epitopes covered 5 out of 10 antigens and 26 out of 47 HLA alleles with a population coverage of 99.66%. On average, each epitope was predicted to bind to 14 ± 3.3 HLA alleles. According to the Immune Epitope Database[209], 10 out of 10 epitopes are known HLA binders or substrings of known binders and 5 out of 10 are T-cell reactive epitopes or substrings of such epitopes (Appendix Table F.6).

7.4 ImmunoNodes - Bringing Immunoinformatics to KNIME

The content of this chapter is part of an unpublished manuscript:

***Schubert, B.**, De la Garza, L., Mohr, C.,* et al. *(2016).*
ImmunoNodes - Graphical Development of Complex Immunoinformatics Workflows

ImmunoNodes is an immunoinformatics toolbox that is fully integrated into a visual workflow development environment called Konstanz Information Miner Analytics Platform (KNIME)[23,24]. KNIME is a free, stand-alone, open-source, workflow development framework for personal computers. Out of the box, it includes hundreds of sample workflows, more than 1,000 nodes with a comprehensive range of solutions for statistics analysis, data acquisition, and visualization. Since it is based on the Eclipse Integrated Development Environment (IDE), is possible to run KNIME on all major platforms. Using the plugin features of Eclipse, it is easy to extend the basic KNIME workbench functionalities by writing extensions, making it the perfect platform for integrative immunoinformatics analysis and pipeline execution within the users' local computing environment.

7.4.1 Implementation

ImmunoNodes provides the same functionality as ETK 2 (Section 7.3) and was also written in Python using FRED 2 (Section 7.2), but is independent of a web service. Being fully integrated into KNIME, immunoinformatics workflows can be executed locally either on a personal computer or on distributed computing systems like clusters or grids. It thus solves technical and legal issues that would otherwise prevent the usage of web-based solutions such as ETK.

ImmunoNodes' KNIME integration was made possible by using the Generic KNIME node (GKN) extension. GKN was developed to assist users to add arbitrary command line tools into KNIME. Each command line tool must provide a description of how to interact with it in the form of Common Tool Descriptor (CTD) file[25]. A CTD file is an XML document that defines the inputs, outputs, and parameters of a command line tool, as well as the excepted parameter types (Listing 7.2). Using such a CTD file, GKN can then automatically create the command line call and execute the program.

Many of the software components used in ImmunoNodes are often difficult to install for untrained individuals and are only available in Unix-based operating systems. To overcome these limitations, we have extended GKN to natively execute command line tools provided within a Docker container. Docker is a software project that enables a lightweight virtualization of software applications, which internally allows an easy deployment of fully configured software suites to the end user. In other words, the burden of installation is

Listing 7.2: Sample CTD file describing the interaction between EpitopePrediction and the end user.

```
<tool name="EpitopePredicton">
  <PARAMETERS>
    <NODE description="Epitope prediction" name="EpitopePredicton">
      <ITEM name="input"   type="input-file"   supported_formats="*.tsv,*.csv"/>
      <ITEM name="alleles" type="input-file"   supported_formats="*.tsv,*.csv"/>
      <ITEM name="output"  type="output-file"  supported_formats="*.tsv,*.csv"/>
      <ITEM name="method"  type="string"       restrictions="netmhc, smmpmbec"/>
      <ITEM name="length"  type="int"          restrictions="8:16" />
   </NODE>
  </PARAMETERS>
  <cli>
    <clielement optionIdentifier="--input">
      <mapping referenceName="EpitopePredicton.input" />
    </clielement>
    <clielement optionIdentifier="--alleles">
      <mapping referenceName="EpitopePredicton.alleles" />
    </clielement>
    <clielement optionIdentifier="--output">
      <mapping referenceName="EpitopePredicton.output" />
    </clielement>
    <clielement optionIdentifier="--method">
      <mapping referenceName="EpitopePredicton.method" />
    </clielement>
    <clielement optionIdentifier="--length">
      <mapping referenceName="EpitopePredicton.length" />
    </clielement>
  </cli>
</tool>
```

shifted from the user to the developer, who has to provide fully configured so-called Docker images. But by far the greatest advantage of Docker is the execution of Linux-restricted software on Windows and Mac OS X operating systems. GKN is now able to automatically generate the required Docker calls and can handle the interaction between the host system and the virtualized Docker container (Figure 7.5).

To enable GKN to perform these tasks, two new classes were introduced, *DockerCommandGenerator* and *LocalDockerToolExecutor*. Once a particular Docker-extended GKN node is executed, *DockerCommandGenerator* identifies input and output files and defines the mount points within the Docker container of that node, generates the command line call by parsing the provided CTD file, and alters the input and output paths to fit the mount points defined within the Docker container. *LocalDockerToolExecutor* is executing the Docker modified command line call, checks whether the Docker daemon is running, and starts the daemon if necessary. In addition to that, we extended the property file, which each GKN project has to provide, to include a specification of which Docker-daemon should be employed, and which Docker image each node of the GKN project should use respectively.

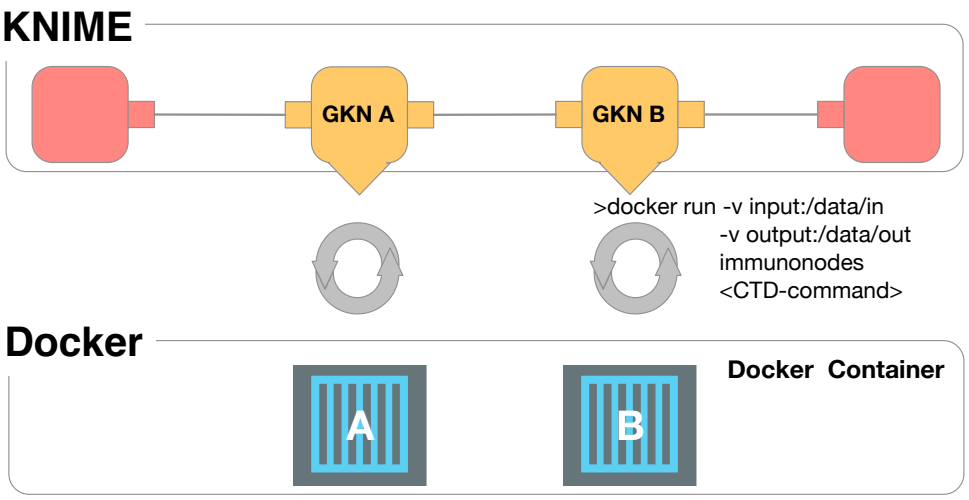


Figure 7.5: Interaction between Generic KNIME Nodes (GKN) and Docker containers. We extended GKN in order to be able to execute command line tools wrapped in a Docker container directly from KNIME. A Docker-extended GKN node first gathers all input and output files to define mount points within the Docker container. Then it generates the standard command line call based on the provided Common Tool Descriptor (CTD) file, while substituting the input and output paths of the host system with the newly generated mount points within the Docker container. Finally, it adds the Docker-specific command line calls, and activates a pre-defined Docker image, which contains the relevant command line tool.

7.4.2 Application

Immunotherapy-based neo-epitopes have become a promising tool in the fight against cancer. Therefore, it is vital to be able to identify potential neo-epitopes quickly, given the tumor-specific mutations of a patient. Immunoinformatics plays a critical role in this effort. To illustrate the usage of ImmunoNodes for neo-epitope identification, we extracted publicly available somatic mutations of a patient (TCGA-A6-2670-01) from the cancer genome atlas that was part of the colon adenocarcinoma cohort[226]. The variants were

Sequence	A*02:01 (nM)	Antigen	Variants (GRCh37)	COSMIC ID
KLFSVFYAV	3.02	OR2F1	g.143657901A>G	265540
KLIQFLMSL	7.98	HSF1	g.145535024C>G	265532
VLTPMLNPM	161.63	OR2F1	g.143657901A>G	265540
SLKDKAWKL	180.10	CYTL1	g.5018601G>A	734242, 265525
LIQFLMSLV	302.70	HSF1	g.145535024C>G	265532

Table 7.1: Predicted neo-epitopes of TCGA-A6-2670-01.

annotated with ANOVAR[211] (2015Dec03), filtered for missense mutations, and used as input into the neo-epitope prediction node. Since the HLA genotype of the patient was not

known, and sequencing data of TCGA for HLA inference with the HLATyping node was restricted, we used HLA-A*02:01 for this illustrative purpose. Five neo-epitopes could be identified falling below a binding threshold of < 500nM (Table 7.1), of which two originated from antigen OR2F1, two from HSF1, and one from CYTL1. OR2F1 and HSF1 were highly expressed in the patient's tumor sample compared to the rest of the cohort (in the 98% percentile and 83% percentile respectively), while CYTL1 was only weakly expressed, making the neo-epitopes that origin from OR2F1 and HSF1 possible vaccine candidates (expression data retrieved from cBioPortal[227]).

7.5 Discussion

Rapid prototyping and development of reliable pipelines is at the heart of any fast-paced research area such as precision medicine. Non-compliant software interfaces and the lack of standardized output formats prolong the implementation process inadvertently, which is particularly the case for growing fields such as computational immunomics, where no consolidation has happened yet. That is why we developed FRED 2, a versatile immunoinformatics software framework enabling a unified interface to many tools, from epitope prediction, HLA typing, to epitope selection and assembly. FRED 2 allows developers to implement novel analysis workflows quickly while maintaining interoperability via well-defined interfaces and output formats. Its openness, intuitive use, and easy extensibility make FRED 2 a perfect hub for advanced immunoinformatics application development, thus constituting a great asset for the future progress of the field. With a growing developer base, the much-needed standardization of interfaces and formats could be established and refined.

The transfer of advanced immunoinformatics applications into a routine usage by clinicians and biologists remains challenging due to often complicated installation procedures and confusing user interfaces with little to no guidance. We therefore developed a web service called EpiToolKit 2 based on the Galaxy platform that enables users to create immunoinformatics workflows by just visually combining simple building blocks without the need of installing any software. Additionally, EpiToolKit 2 provides extensive guidance in each configuration step of the individual components. It also allows the user to store, version control, and share the developed workflows together with the used tool settings increasing reproducibility. Beyond the presented application, EpiToolKit 2 can be used to tackle a manifold of other immunological questions and thus should not only be valuable for applied but also for basic immunological research.

The unique nature of biomedical research, however, often restricts the usage of web-based solutions due to data volumes or legal issues. Having these aspects in mind, we developed ImmunoNodes, an immunoinformatics KNIME plug-in for desktop computers

that has the same functionality as EpitToolKit 2 and maintains all benefits of Galaxy-based web service. Due to our newly developed GKN extension, which enables the call of dockerized software from within KNIME, the complexity of installation and configuration of required third-party libraries has been lifted from the end user as a result of the provided Docker images. The newly developed capability of GKN of calling arbitrary dockerized command line tools opens up a broad variety of bioinformatics applications that could be integrated into KNIME with minimal effort. Several initiatives, such as BioDocker (`http://biodocker.org/`), BioBox (`http://bioboxes.org/`), and BioShaDock[228] have already begun to compile pre-configured bioinformatics software that could potentially be integrated into KNIME using the extended GKN making KNIME to one of the most attractive platforms for bioinformatics practitioners. It would allow quick and straightforward implementations of complex workflows needed in, for example, multi-omics studies or immunotherapy development.

To summarize, we developed software solutions that enable bioinformatics developers to implement novel immunoinformatics pipelines quickly while sustaining interchangeability of methods due to unified interfaces and output formats. We also established a web-based and a desktop solution for practitioners to build immunoinformatics analysis workflows using a graphical interface with pre-defined building blocks facilitating the daily use of advanced immunoinformatics methods in biomedical research.

Chapter 8

Conclusion and Outlook

Precision and personalized medicine is a revolutionizing step in health care practice that tailors the treatment decision and the development of new drugs to fit the genetic prerequisites of a specific sub-population or even an individual patient, thereby increasing the efficacy of the treatment while decreasing its side effects. Its main application so far has been in oncology, where the immune system plays a vital role for a successful treatment. The subtle immunological and genetic differences between patients can have a large effect on treatment outcome. Thus, it is important to address the immune system's heterogeneity, in particular that of the HLA gene cluster, when treating a patient with cancer or other diseases where the immune system is involved (e.g., autoimmune diseases or infectious diseases).

In the first part of this thesis, we presented a new and highly efficient method to identify an individual's HLA genotype based on standard NGS sequencing data (Chapter 4). In contrast to traditional methods that require the creation of additional data to elucidate the HLA genotype, this method can directly utilize pre-existing sequencing data used for other diagnostic purposes and thus reduce cost and time. Especially in oncology, where sequencing-based diagnostics have been widely implemented and will become part of the standard care in the near future, algorithmic-based solutions pose a cost-effective alternative to traditional HLA genotyping. However, there are several shortcomings of existing methods. Most algorithmic solutions do not achieve the needed accuracy for clinical applications, although a recent comparative study indicates sufficient performance of OptiType on high quality data[87]. This should be further explored to (a) identify quality criteria of biomedically used sequencing data and (b) to thoroughly re-evaluate the genotyping performance of OptiType on such high quality data. On a related note, the prediction performance of current HLA typing algorithms is tightly linked to the quality of the underlying reference database. An immediate quality increase could be achieved by fully characterizing the partly sequenced HLA alleles. Also, OptiType currently lacks

a quality measure of its prediction, which is vital for clinical applications. One possible indicator could be the skewness of the top α-percent solutions. The distance between the top solution and the next best solutions should reflect the uncertainty of the inferred genotype under the premise that the solution ought to explain the majority of reads. A more radical approach would be the re-formulation of OptiType's model in a Bayesian setting using a sparsity inducing hierarchical model with a group or set-cover prior[229].

We then discussed two scenarios, in which a stratification based on a populations' HLA distribution or on an individuals' HLA genotype is necessary to improve treatment outcome while reducing adverse effects. In the first example (Chapter 5), we developed an advanced approach to design so-called string-of-beads vaccines, that are polypeptide or RNA/DNA vaccines composed of concatenated epitopes. The approach maximizes the recovery likelihood of the contained therapeutic peptides in order to increase vaccine efficacy. The recovery is influenced by the ordering of the epitopes and the connecting spacer sequences between epitopes. HLA stratification was used to reduce the risk of unwanted immune reactions by newly arising artificial peptides that can bind to HLA molecules after miscleavage events. We compared the *in silico* designed string-of-beads vaccines with experimentally tested vaccines and showed that the experimental spacer sequences used in the literature are often sub-optimally chosen. Together with collaborators, we are now working on an experimental study to substantiate the method's efficacy. To this end, we designed pairs of string-of-beads vaccines with highest and lowest recovery probability based on prior extracted epitopes with identified reactive T cell clones. In such a setting, it is possible to directly identify the presented epitopes and thus measure the algorithm's influence on epitope recovery by comparing the predicted worst performing with the predicted best performing string-of-beads construct. A general problem of the proposed method is its reliance on existing cleavage prediction methods, which have been far from the prediction accuracy of HLA binding methods. The development of more accurate approaches is mainly hampered by the lack of high quality data[121]. With newly developed mass spectrometry-based HLA-ligand identification methods[230], large, high-quality data will be available in the near future that can be utilized to develop advanced cleavage prediction methods. These data, however, are generated by a mixture of various cleavage events, from (immuno)proteasomal cleavage to alternative pathways. Thus, methods seeking to use these data have to account for its heterogeneity to optimally capitalize on the wealth of data.

While we have taken an iterative approach to vaccine design, by relying upon the identification of a set of therapeutically usable epitopes before assembly, both selection and assembly methods could be combined using integer linear programming. To this end, two combinatorial problems, the traveling salesman problem and the subset selection problem, have to be jointly solved. This class of problems is often encountered in operations research under the name of *orienteering* or *selective traveling salesman problem*[231]. Such

an orienteering model could also be used to design so-called mosaic vaccines by using a peptide overlapping graph[232] as the underlying data structure on which the traveling salesman instance is defined. A mosaic vaccine is a polypeptide constructed by highly overlapping peptide sequences and is a largely unexplored class of vaccines with primary application in vaccine development against highly polymorphic viruses, such as influenza and HIV[233–235]. Mosaic vaccines possess the benefit of incorporating a larger proportion of therapeutic epitopes than string-of-bead vaccines, and thus can cover a substantial fraction of virus and HLA variability at once. However, such models are very hard to solve and need sophisticated approximation algorithms to tackle realistic problem sizes.

In the second example (Chapter 6), we discussed the issue of anti-drug antibody (ADA) formation of biotherapeutics that causes a reduction of therapeutic efficacy or even systemic allergic reactions. We also introduced an experimental procedure - termed de-immunization - that targets ADA causing epitopes presented on HLA-II molecules by sequence alteration. Such a procedure is highly time- and resource consuming and can only explore low amounts of mutations (in the order of 1-3) simultaneously. We therefore developed a computational approach that finds immunogenicity-reducing variants without disrupting the functional and structural integrity of the protein strongly. For this purpose, we utilized a recently developed sequence-based statistical model for structural *ab initio* and variant effect prediction and combined it with the immunogenicity function developed in Chapter 5 in a bi-objective mixed integer linear programming framework. We then applied our mathematical model to de-immunize the C2 domain of factor VIII, which is linked to ADA formation in hemophilia A patients treated with factor VIII. Our subsequent experimental analysis of the found mutations confirmed our underlying assumptions and yielded several immunogenicity-reducing candidates for further study.

Similar ideas could be applied to antibody humanization. The sequence-based statistical model could be used to identify suitable human antibody templates for a given murine monoclonal antibody. To do so, the two-dimensional Hamiltonian has to be condensed to a one-dimensional measure that can be utilized for sequence comparison. In theory, this condensed sequence profile can be more expressive than standard sequence similarity-based matrices, as it not only encodes evolutionary but also structural and functional information. Generally, Potts models could potentially replace hidden Markov models for sequence profiling and similar tasks where hidden Markov models are standardly employed.

In conjunction with this study, we developed a new and highly parallelized bi-objective integer linear programming solver to be able to handle the inherent multiobjective nature of the de-immunization problem. Our computational analysis showed its excellent runtime behavior compared to other conventional approaches with a 25-fold speedup compared to one of the best criterion search algorithms, the balanced box algorithm. The solver is generically applicable to all bi-objective mixed integer linear programs, where only one

objective exhibits the mixed integer property. Thus, the solver will be useful in other areas, since freely available implementations of multiobjective solvers and especially of distributed solvers, are rare. Nevertheless, solving a de-immunization problem of considerable size takes a substantial amount of time due to the form of the protein fitness function, as its has been shown to be NP-hard[236] and inapproximable[237]. Exact solutions, therefore, can only be obtained for medium-sized problem instances. To overcome these limitations, approximate solving schemes based on belief propagation[238,239] could be used by reparameterizing the likelihood to incorporate the second objective function. Solving such a model iteratively with different values of the mixing hyperparameter of the two objectives would yield approximate non-dominated points that reside on the convex hull of the true Pareto front.

Both presented applications use an abstract approximation of immunogenicity based on HLA binding affinity. Despite the progress immunoinformatics has made in the last decades, the prediction of T-cell epitopes (i.e., epitopes that elicit a T-cell response) is still an unsolved problem. The development of such prediction methods will be a major effort in the near future, and will have great impact on vaccine design and other immunomodulatory therapies where an immune reaction should be invoked. In the presented applications, however, using HLA binding as a proxy of immunogenicity suffices, as the goal is to suppress an immune response and hindering HLA binding is enough to do so.

In general, it is difficult to measure an immune reaction, as multiple entities are involved and no single well-characterized indicator exists. Especially the dynamics of an immune reaction during treatment are not well understood. It will thus become important in the near future to develop reliable methods to adequately describe the entirety of the immune reaction. This entails the identification and quantification of involved immunity related cells (including cells of the innate immune system), as well as the prevalent T and B cell clonal population by using sequencing technologies or other means. Such methods will have immediate effect on applied oncological research as the standard care, as well as experimental treatments are not well understood in terms of their immunological influence. Once the immunological dynamics are well characterized, rational combination therapies can be devised and optimal vaccination schedules can be established, which will help to overcome the problem of tumor resistance.

In the second part of the thesis (Chapter 7), we developed programming libraries and software solutions to enable a rapid development and daily use of advanced immunoinformatics applications in biomedical research. In particular, we implemented FRED 2, a Python framework for immunoinformatics applications that offers developers unified interfaces and output formats for various immunoinformatics tools, as well as necessary pre- and post-processing procedures. It allows examining different prediction tools without losing interoperability and without the need of developing custom input and output parsers. Due to

FRED 2's open nature and well-defined interfaces, it could become an immunoinformatics development hub as OpenMS[240] has become in the field of mass spectrometry.

Based on FRED 2, we developed two software applications that liberate biomedical researchers to craft their own immunoinformatics analysis workflows without the need of knowing a programming language by using visual workflow managers such as Galaxy and KNIME. In the case of the latter, we extended its capability via the generic KNIME node extension to interact with Docker, a lightweight virtualization engine that allows for the pre-packing and executing of software independent of the users operation system (OS). This extension will ease the development of future KNIME extensions, as developers do not have to account for OS dependencies and peculiarities. It also enables the KNIME community to exploit already existing Docker images to increase KNIME's functionality tremendously with little effort.

To summarize, the presented methods represent a small contribution to precision medicine and will be of interest to basic immunological research, oncology, and applied biomedical research. It also could have a substantial impact on vaccine as well as biotherapeutic development. One of the remaining challenges is the translation of these advanced computer-aided approaches into clinical practice. Providing intuitively usable web-applications or graphical programming front ends, like the ones presented here, constitute such a possibility.

Bibliography

[1] Bollag, G., Hirth, P., Tsai, J., Zhang, J., Ibrahim, P.N., Cho, H., Spevak, W., Zhang, C., Zhang, Y., Habets, G., *et al.*: Clinical efficacy of a RAF inhibitor needs broad target blockade in BRAF-mutant melanoma. Nature **467**(7315), 596–599 (2010) 1

[2] Hatzivassiliou, G., Song, K., Yen, I., Brandhuber, B.J., Anderson, D.J., Alvarado, R., Ludlam, M.J., Stokoe, D., Gloor, S.L., Vigers, G., *et al.*: RAF inhibitors prime wild-type RAF to activate the MAPK pathway and enhance growth. Nature **464**(7287), 431–435 (2010) 1

[3] Halaban, R., Zhang, W., Bacchiocchi, A., Cheng, E., Parisi, F., Ariyan, S., Krauthammer, M., McCusker, J.P., Kluger, Y., Sznol, M.: PLX4032, a selective BRAFV600E kinase inhibitor, activates the ERK pathway and enhances cell migration and proliferation of BRAFWT melanoma cells. Pigment cell & melanoma research **23**(2), 190–200 (2010) 1

[4] Peters, S., Michielin, O., Zimmermann, S.: Dramatic response induced by vemurafenib in a BRAF V600E-mutated lung adenocarcinoma. Journal of Clinical Oncology **31**(20), 341–344 (2013) 1

[5] Zhu, X., Petrovski, S., Xie, P., Ruzzo, E.K., Lu, Y.-F., McSweeney, K.M., Ben-Zeev, B., Nissenkorn, A., Anikster, Y., Oz-Levi, D., et al.: Whole-exome sequencing in undiagnosed genetic diseases: interpreting 119 trios. Genetics in Medicine (2015) 1

[6] Solberg, O.D., Mack, S.J., Lancaster, A.K., Single, R.M., Tsai, Y., Sanchez-Mazas, A., Thomson, G.: Balancing selection and heterogeneity across the classical human leukocyte antigen loci: a meta-analytic review of 497 population studies. Human immunology **69**(7), 443–464 (2008) 2

[7] 1000 Genomes Project Consortium and others: A global reference for human genetic variation. Nature **526**(7571), 68–74 (2015) 3

[8] Weinstein, J.N., Collisson, E.A., Mills, G.B., Shaw, K.R.M., Ozenberger, B.A., Ellrott, K., Shmulevich, I., Sander, C., Stuart, J.M., Network, C.G.A.R., *et al.*: The cancer genome atlas pan-cancer analysis project. Nature genetics **45**(10), 1113–1120 (2013) 3

[9] Gubin, M.M., Zhang, X., Schuster, H., Caron, E., Ward, J.P., Noguchi, T., Ivanova, Y., Hundal, J., Arthur, C.D., Krebber, W.-J., *et al.*: Checkpoint blockade cancer immunotherapy targets tumour-specific mutant antigens. Nature **515**(7528), 577–581 (2014) 3

[10] Yadav, M., Jhunjhunwala, S., Phung, Q.T., Lupardus, P., Tanguay, J., Bumbaca, S., Franci, C., Cheung, T.K., Fritsche, J., Weinschenk, T., *et al.*: Predicting immunogenic tumour mutations by combining mass spectrometry and exome sequencing. Nature **515**(7528), 572–576 (2014) 3

[11] Boisguérin, V., Castle, J., Loewer, M., Diekmann, J., Mueller, F., Britten, C., Kreiter, S., Türeci, Ö., Sahin, U.: Translation of genomics-guided RNA-based personalised cancer vaccines: towards the bedside. British journal of cancer **111**(8), 1469–1475 (2014) 3, 85

[12] Tran, E., Turcotte, S., Gros, A., Robbins, P.F., Lu, Y.-C., Dudley, M.E., Wunderlich, J.R., Somerville, R.P., Hogan, K., Hinrichs, C.S., *et al.*: Cancer immunotherapy based on mutation-specific CD4+ T cells in a patient with epithelial cancer. Science **344**(6184), 641–645 (2014) 3

[13] Ding, F.-X., Wang, F., Lu, Y.-M., Li, K., Wang, K.-H., He, X.-W., Sun, S.-H.: Multiepitope peptide-loaded virus-like particles as a vaccine against hepatitis B virus–related hepatocellular carcinoma. Hepatology **49**(5), 1492–1502 (2009) 3, 45, 55, 57

[14] Cornet, S., Miconnet, I., Menez, J., Lemonnier, F., Kosmatopoulos, K.: Optimal organization of a polypeptide-based candidate cancer vaccine composed of cryptic tumor peptides with enhanced immunogenicity. Vaccine **24**(12), 2102–2109 (2006) 3, 45

[15] Velders, M.P., Weijzen, S., Eiben, G.L., Elmishad, A.G., Kloetzel, P.-M., Higgins, T., Ciccarelli, R.B., Evans, M., Man, S., Smith, L., *et al.*: Defined flanking spacers and enhanced proteolysis is essential for eradication of established tumors by an epitope string DNA vaccine. The Journal of Immunology **166**(9), 5366–5373 (2001) 3, 45, 53, 55

[16] Kreiter, S., Vormehr, M., van de Roemer, N., Diken, M., Löwer, M., Diekmann, J., Boegel, S., Schrörs, B., Vascotto, F., Castle, J.C., *et al.*: Mutant MHC class II epitopes drive therapeutic immune responses to cancer. Nature **520**(7549), 692–696 (2015) 3, 45

[17] Schellekens, H.: Bioequivalence and the immunogenicity of biopharmaceuticals. Nature reviews Drug discovery **1**(6), 457–462 (2002) 4, 59, 61

[18] Warmerdam, P.A., Plaisance, S., Vanderlick, K., Vandervoort, P., Brepoels, K., Collen, D., De Maeyer, M., *et al.*: Elimination of a human T-cell region in staphylokinase by T-cell screening and computer modeling. Thrombosis and haemostasis **87**(4), 666–673 (2002) 4, 62

[19] Jones, T., Phillips, W., Smith, B., Bamford, C., Nayee, P., Baglin, T., Gaston, J., Baker, M.: Identification and removal of a promiscuous CD4+ T cell epitope from the C1 domain of factor VIII. Journal of Thrombosis and Haemostasis **3**(5), 991–1000 (2005)

[20] Harding, F.A., Liu, A.D., Stickler, M., Razo, O.J., Chin, R., Faravashi, N., Viola, W., Graycar, T., Yeung, V.P., Aehle, W., *et al.*: A β-lactamase with reduced immunogenicity for the targeted delivery of chemotherapeutics using antibody-directed enzyme prodrug therapy. Molecular cancer therapeutics **4**(11), 1791–1800 (2005) 4

[21] Parker, A.S., Zheng, W., Griswold, K.E., Bailey-Kellogg, C.: Optimization algorithms for functional deimmunization of therapeutic proteins. BMC bioinformatics **11**(1), 180 (2010) 4, 62, 83

[22] Afgan, E., Baker, D., van den Beek, M., Blankenberg, D., Bouvier, D., Čech, M., Chilton, J., Clements, D., Coraor, N., Eberhard, C., et al.: The Galaxy platform for accessible, reproducible and collaborative biomedical analyses: 2016 update. Nucleic acids research, 343 (2016) 5

[23] Berthold, M.R., Cebron, N., Dill, F., Gabriel, T.R., Kötter, T., Meinl, T., Ohl, P., Sieb, C., Thiel, K., Wiswedel, B.: KNIME: The Konstanz information miner. Springer, Heidelberg (2008) 5, 86, 94

[24] Berthold, M.R., Cebron, N., Dill, F., Gabriel, T.R., Kötter, T., Meinl, T., Ohl, P., Thiel, K., Wiswedel, B.: KNIME-the Konstanz information miner: version 2.0 and beyond. AcM SIGKDD explorations Newsletter **11**(1), 26–31 (2009) 5, 86, 94

[25] de la Garza, L., Veit, J., Szolek, A., Röttig, M., Aiche, S., Gesing, S., Reinert, K., Kohlbacher, O.: From the desktop to the grid: scalable bioinformatics via workflow conversion. BMC Bioinformatics **17**(1), 1 (2016) 5, 94

[26] Kindt, T.J., Goldsby, R.A., Osborne, B.A.: Kuby immunology. 6th. WH Freeman., New York (2007) 7

[27] Murphy, K.M.: Janeway's immunobiology. Garland Science, New York (2011) 7

[28] Curiel, T.J.: Tregs and rethinking cancer immunotherapy. The Journal of clinical investigation **117**(5), 1167–1174 (2007) 10

[29] Sette, A., Buus, S., Colon, S., Smith, J.A., Miles, C., Grey, H.M.: Structural characteristics of an antigen required for its interaction with Ia and recognition by T cells (1987) 11

[30] Forman, S.J., Blume, K.G., Thomas, E.D.: Hematopoietic cell transplantation. Blackwell Science, Malden, MA (1999) 11

[31] Geraghty, D., Holdsworth, R., Hurley, C., Lau, M., Lee, K., Mach, B., Maiers, M., Mayr, W., Muller, C., Parham, P., *et al.*: Nomenclature for factors of the HLA system, 2010. Tissue Antigens **75**, 291–455 (2010) 11, 12

[32] Robinson, J., Halliwell, J.A., Hayhurst, J.D., Flicek, P., Parham, P., Marsh, S.G.: The IPD and IMGT/HLA database: allele variant databases. Nucleic Acids Research **43**, 423–4331 (2015) 11, 31, 33

[33] Leone, P., Shin, E.-C., Perosa, F., Vacca, A., Dammacco, F., Racanelli, V.: MHC class I antigen processing and presenting machinery: organization, function, and defects in tumor cells. Journal of the National Cancer Institute, 184 (2013) 12, 13

[34] Heinemeyer, W., Fischer, M., Krimmer, T., Stachon, U., Wolf, D.H.: The active sites of the eukaryotic 20 S proteasome and their involvement in subunit precursor processing. Journal of Biological Chemistry **272**(40), 25200–25209 (1997) 13

[35] Chaturvedi, P., Hengeveld, R., Zechel, M.A., Lee-Chan, E., Singh, B.: The functional role of class II-associated invariant chain peptide (CLIP) in its ability to variably modulate immune responses. International immunology **12**(6), 757–765 (2000) 13

[36] Poluektov, Y.O., Kim, A., Sadegh-Nasseri, S.: HLA-DO and its role in MHC class II antigen presentation. Front Immunol **4**(260.10), 3389 (2013) 13

[37] Pelanda, R., Torres, R.M.: Central B-cell tolerance: where selection begins. Cold Spring Harbor perspectives in biology **4**(4), 007146 (2012) 13

[38] Wolsey, L.A., Nemhauser, G.L.: Integer and combinatorial optimization. John Wiley & Sons, New York (2014) 17

[39] Ehrgott, M.: Multicriteria optimization. Springer, Heidelberg (2006) 17, 18, 24, 25, 27, 47

[40] Dantzig, G.B.: Reminiscences about the origins of linear programming. Springer, Heidelberg (1983) 18

[41] Schrijver, A.: Theory of linear and integer programming. John Wiley & Sons, Chichester (1998) 18, 21

[42] Borgwardt, K.: The Simplex Method: A Probabilistic Analysis, Algorithms and Combinatorics: Study and Research Texts, Vol. l. Springer, Berlin, New York (1987) 18

[43] Nocedal, J., Wright, S.: Numerical optimization. Springer, New York (2006) 19

[44] Cormen, T.H.: Introduction to algorithms. MIT press, Cambridge, MA (2009) 19, 21

[45] Land, A.H., Doig, A.G.: An automatic method of solving discrete programming problems. Econometrica: Journal of the Econometric Society, 497–520 (1960) 22

[46] Dantzig, G., Fulkerson, R., Johnson, S.: Solution of a large-scale traveling-salesman problem. Journal of the operations research society of America **2**(4), 393–410 (1954) 22

[47] Gomory, R.E.: Outline of an algorithm for integer solutions to linear programs and an algorithm for the mixed integer problem. In: 50 Years of Integer Programming 1958-2008, pp. 77–103. Springer, New York (2010) 22

[48] Balas, E., Ceria, S., Cornuéjols, G., Natraj, N.: Gomory cuts revisited. Operations Research Letters **19**(1), 1–9 (1996) 24

[49] Boland, N., Charkhgard, H., Savelsbergh, M.: A criterion space search algorithm for biobjective integer programming: The balanced box method. INFORMS Journal on Computing **27**(4), 735–754 (2015) 25, 68, 69, 71, 73

[50] Isermann, H.: The enumeration of the set of all efficient solutions for a linear multiple objective program. Operational Research Quarterly, 711–725 (1977) 25

[51] Vira, C., Haimes, Y.Y.: Multiobjective decision making: theory and methodology vol. 8. North-Holland, New York (1983) 25

[52] Guddat, J.: Multiobjective and stochastic optimization based on parametric optimization vol. 26. Akademie-Verlag, Berlin (1985) 26

[53] Ehrgott, M., Ryan, D.M.: Constructing robust crew schedules with bicriteria optimization. Journal of Multicriteria Decision Analysis **11**(3), 139 (2002) 26

[54] Bowman Jr, V.J.: On the relationship of the Tchebycheff norm and the efficient frontier of multiple-criteria objectives. In: Multiple Criteria Decision Making, pp. 76–86. Springer, Heidelberg (1976) 26

[55] Steuer, R.E., Choo, E.-U.: An interactive weighted Tchebycheff procedure for multiple objective programming. Mathematical programming **26**(3), 326–344 (1983)

[56] Ralphs, T.K., Saltzman, M.J., Wiecek, M.M.: An improved algorithm for solving biobjective integer programs. Annals of Operations Research **147**(1), 43–70 (2006) 26

[57] Ovsyannikova, I.G., Poland, G.A.: Vaccinomics: current findings, challenges and novel approaches for vaccine development. The AAPS journal **13**(3), 438–444 (2011) 29

[58] Haralambieva, I.H., Ovsyannikova, I.G., Pankratz, V.S., Kennedy, R.B., Jacobson, R.M., Poland, G.A.: The genetic basis for interindividual immune response variation to measles vaccine: new understanding and new vaccine approaches. Expert review of vaccines. **12**(1), 57–70 (2013) 29

[59] Bradley, B.: The role of HLA matching in transplantation. Immunology letters **29**(1), 55–59 (1991) 29

[60] Mytilineos, J., Opelz, G., Wujciak, T., *et al.*: HLA compatibility and organ transplant survival. Rev Immunogenet **1**, 334 (1999) 29

[61] Undlien, D.E., Lie, B.A., Thorsby, E.: HLA complex genes in type 1 diabetes and other autoimmune diseases. Which genes are involved? Trends in genetics **17**(2), 93–100 (2001) 29

[62] Thorsby, E., Lie, B.A.: HLA associated genetic predisposition to autoimmune diseases: Genes involved and possible mechanisms. Transplant immunology **14**(3), 175–182 (2005) 29

[63] Erlich, H.: HLA DNA typing: past, present, and future. Tissue Antigens **80**(1), 1–11 (2012) 29

[64] Gabriel, C., Danzer, M., Hackl, C., Kopal, G., Hufnagl, P., Hofer, K., Polin, H., Stabentheiner, S., Pröll, J.: Rapid high-throughput human leukocyte antigen typing by massively parallel pyrosequencing for high-resolution allele identification. Hum Immunol **70**(11), 960–964 (2009) 30

[65] Lank, S.M., Wiseman, R.W., Dudley, D.M., O'Connor, D.H.: A novel single cDNA amplicon pyrosequencing method for high-throughput, cost-effective sequence-based HLA class I genotyping. Human immunology **71**(10), 1011–1017 (2010)

[66] Moonsamy, P., Williams, T., Bonella, P., Holcomb, C., Höglund, B., Hillman, G., Goodridge, D., Turenchalk, G., Blake, L., Daigle, D., *et al.*: High throughput HLA genotyping using 454 sequencing and the Fluidigm Access Array™system for simplified amplicon library preparation. Tissue antigens **81**(3), 141–149 (2013)

[67] Danzer, M., Niklas, N., Stabentheiner, S., Hofer, K., Pröll, J., Stückler, C., Raml, E., Polin, H., Gabriel, C.: Rapid, scalable and highly automated HLA genotyping using next-generation sequencing: a transition from research to diagnostics. BMC genomics **14**(1), 221 (2013) 30

[68] Erlich, R.L., Jia, X., Anderson, S., Banks, E., Gao, X., Carrington, M., Gupta, N., DePristo, M.A., Henn, M.R., Lennon, N.J., *et al.*: Next-generation sequencing for HLA typing of class I loci. BMC genomics **12**(1), 42 (2011) 30, 39, 40

[69] Warren, R.L., Choe, G., Freeman, D.J., Castellarin, M., Munro, S., Moore, R., Holt, R.A.: Derivation of HLA types from shotgun sequence datasets. Genome Med **4**(12), 95 (2012) 30, 36

[70] Boegel, S., Lower, M., Schafer, M., Bukur, T., De Graaf, J., Boisguérin, V., Tureci, O., Diken, M., Castle, J.C., Sahin, U.: HLA typing from RNA-Seq sequence reads. Genome Med **4**(12), 102 (2013) 31, 36, 140

[71] Liu, C., Yang, X., Duffy, B., Mohanakumar, T., Mitra, R.D., Zody, M.C., Pfeifer, J.D.: ATHLATES: accurate typing of human leukocyte antigen through exome sequencing. Nucleic acids research **41**(14), 142–142 (2013) 31, 36, 140

[72] Kim, H.J., Pourmand, N.: HLA haplotyping from RNA-seq data using hierarchical read weighting. Plos One **8**(e67885) (2013) 31, 36, 39

[73] Major, E., Rigo, K., Hague, T., Berces, A., Juhos, S.: HLA typing from 1000 genomes whole genome and whole exome illumina data. Plos One **8**(e78410) (2013) 31, 36

[74] Blasczyk, R., Kotsch, K., Wehling, J.: The Nature of Polymorphism of the HLA Class I Non-Coding Regions and Their Contribution to the Diversification of HLA. Hereditas **127**(1-2), 7–9 (1997) 31

[75] Wilbur, W.J., Lipman, D.J.: Rapid similarity searches of nucleic acid and protein data banks. Proceedings of the National Academy of Sciences **80**(3), 726–730 (1983) 33

[76] Sievers, F., Wilm, A., Dineen, D., Gibson, T.J., Karplus, K., Li, W., Lopez, R., McWilliam, H., Remmert, M., Söding, J., *et al.*: Fast, scalable generation of high-quality protein multiple sequence alignments using Clustal Omega. Molecular systems biology **7**(1), 539 (2011) 33

[77] Szolek, A., Schubert, B., Mohr, C., Sturm, M., Feldhahn, M., Kohlbacher, O.: OptiType: precision HLA typing from next-generation sequencing data. Bioinformatics **30**(23), 3310–3316 (2014) 33, 140

[78] Weese, D., Holtgrewe, M., Reinert, K.: RazerS 3: faster, fully sensitive read mapping. Bioinformatics **28**(20), 2592–2599 (2012) 33

[79] Gonzalez-Galarza, F.F., Christmas, S., Middleton, D., Jones, A.R.: Allele frequency net: a database and online repository for immune gene frequencies in worldwide populations. Nucleic acids research **39**(suppl 1), 913–919 (2011) 33

[80] NCBI Resource Coordinators: Database resources of the National Center for Biotechnology Information. Nucleic acids research **41**(Database issue), 8 (2013) 33, 36

[81] International HapMap Consortium and others: A haplotype map of the human genome. Nature **437**(7063), 1299–1320 (2005) 36

[82] Montgomery, S.B., Sammeth, M., Gutierrez-Arcelus, M., Lach, R.P., Ingle, C., Nisbett, J., Guigo, R., Dermitzakis, E.T.: Transcriptome genetics using second generation sequencing in a Caucasian population. Nature **464**(7289), 773–777 (2010) 36

[83] 1000 Genomes Project Consortium and others: An integrated map of genetic variation from 1,092 human genomes. Nature **491**(7422), 56–65 (2012) 36

[84] Weese, D., Holtgrewe, M., Reinert, K.: RazerS 3: faster, fully sensitive read mapping. Bioinformatics **28**(20), 2592–2599 (2012) 36

[85] Langmead, B., Salzberg, S.L.: Fast gapped-read alignment with Bowtie 2. Nature methods **9**(4), 357–359 (2012) 36

[86] Hart, W.E., Laird, C., Watson, J.-P., Woodruff, D.L.: Pyomo–optimization modeling in python vol. 67. Springer, New York (2012) 36

[87] Bauer, D.C., Zadoorian, A., Wilson, L.O., Thorne, N.P., et al.: Evaluation of computational programs to predict HLA genotypes from genomic sequencing data. Briefings in Bioinformatics, 097 (2016) 38, 40, 99

[88] Shukla, S.A., Rooney, M.S., Rajasagi, M., Tiao, G., Dixon, P.M., Lawrence, M.S., Stevens, J., Lane, W.J., Dellagatta, J.L., Steelman, S., *et al.*: Comprehensive analysis of cancer-associated somatic mutations in class I HLA genes. Nature biotechnology **33**(11), 1152–1158 (2015) 40, 85, 140

[89] Patro, R., Mount, S.M., Kingsford, C.: Sailfish enables alignment-free isoform quantification from RNA-seq reads using lightweight algorithms. Nature biotechnology **32**(5), 462–464 (2014) 41

[90] Bray, N.L., Pimentel, H., Melsted, P., Pachter, L.: Near-optimal probabilistic RNA-seq quantification. Nature biotechnology **34**(5), 525–527 (2016) 41

[91] de Quadros, C.A.: History and prospects for viral disease eradication. Medical microbiology and immunology **191**(2), 75–81 (2002) 43

[92] Provost, P., Hughes, J., Miller, W., Giesa, P., Banker, F., Emini, E.: An inactivated hepatitis A viral vaccine of cell culture origin. Journal of medical virology **19**(1), 23–31 (1986) 43

[93] Oxman, M., Levin, M., Johnson, G., Schmader, K., Straus, S., Gelb, L., Arbeit, R., Simberkoff, M., Gershon, A., Davis, L., *et al.*: A vaccine to prevent herpes zoster and postherpetic neuralgia in older adults. New England Journal of Medicine **352**(22), 2271–2284 (2005)

[94] Goepfert, P.A., Tomaras, G.D., Horton, H., Montefiori, D., Ferrari, G., Deers, M., Voss, G., Koutsoukos, M., Pedneault, L., Vandepapeliere, P., *et al.*: Durable HIV-1 antibody and T-cell responses elicited by an adjuvanted multi-protein recombinant vaccine in uninfected human volunteers. Vaccine **25**(3), 510–518 (2007)

[95] Mancini-Bourgine, M., Fontaine, H., Bréchot, C., Pol, S., Michel, M.-L.: Immunogenicity of a hepatitis B DNA vaccine administered to chronic HBV carriers. Vaccine **24**(21), 4482–4489 (2006) 43

[96] Purcell, A.W., McCluskey, J., Rossjohn, J.: More than one reason to rethink the use of peptides in vaccine design. Nature reviews Drug discovery **6**(5), 404–414 (2007) 43, 44

[97] Schubert, B., Lund, O., Nielsen, M.: Evaluation of peptide selection approaches for epitope-based vaccine design. Tissue antigens **82**(4), 243–251 (2013) 44

[98] Sette, A., Fikes, J.: Epitope-based vaccines: an update on epitope identification, vaccine design and delivery. Current opinion in immunology **15**(4), 461–470 (2003) 44

[99] Livingston, B.D., Newman, M., Crimi, C., McKinney, D., Chesnut, R., Sette, A.: Optimization of epitope processing enhances immunogenicity of multiepitope DNA vaccines. Vaccine **19**(32), 4652–4660 (2001) 45

[100] Vider-Shalit, T., Raffaeli, S., Louzoun, Y.: Virus-epitope vaccine design: informatic matching the HLA-I polymorphism to the virus genome. Molecular immunology **44**(6), 1253–1261 (2007) 46

[101] Toussaint, N.C., Maman, Y., Kohlbacher, O., Louzoun, Y.: Universal peptide vaccines - Optimal peptide vaccine design based on viral sequence conservation. Vaccine **29**(47), 8745–8753 (2011) 46, 87, 91, 140

[102] Antonets, D.V., Bazhan, S.I.: PolyCTLDesigner: a computational tool for constructing polyepitope T-cell antigens. BMC research notes **6**(1), 407 (2013) 46

[103] Köppe, M.: On the complexity of nonlinear mixed-integer optimization. In: Mixed Integer Nonlinear Programming, pp. 533–557. Springer, New York (2012) 47

[104] Hemmecke, R., Köppe, M., Lee, J., Weismantel, R.: Nonlinear integer programming. In: 50 Years of Integer Programming 1958-2008, pp. 561–618. Springer, Berlin, London (2010) 47

[105] Toussaint, N.C., Dönnes, P., Kohlbacher, O.: A mathematical framework for the selection of an optimal set of peptides for epitope-based vaccines. PLoS Comput Biol **4**(12), 1000246 (2008) 48, 87, 91, 92

[106] Sette, A., Vitiello, A., Reherman, B., Fowler, P., Nayersina, R., Kast, W.M., Melief, C., Oseroff, C., Yuan, L., Ruppert, J., *et al.*: The relationship between class I binding affinity and immunogenicity of potential cytotoxic T cell epitopes. The Journal of Immunology **153**(12), 5586–5592 (1994) 48

[107] Orman, A., Williams, H.: A survey of different integer programming formulations of the travelling salesman problem. In: Optimisation, Econometric and Financial Analysis, pp. 91–104. Springer, London (2007) 51

[108] Lin, S., Kernighan, B.W.: An effective heuristic algorithm for the traveling-salesman problem. Operations research **21**(2), 498–516 (1973) 51

[109] Helsgaun, K.: General k-opt submoves for the Lin–Kernighan TSP heuristic. Mathematical Programming Computation **1**(2-3), 119–163 (2009) 51, 52

[110] Helsgaun, K.: An effective implementation of the Lin–Kernighan traveling salesman heuristic. European Journal of Operational Research **126**(1), 106–130 (2000) 52

[111] Rammensee, H.-G., Bachmann, J., Emmerich, N.P.N., Bachor, O.A., Stevanović, S.: SYFPEITHI: database for MHC ligands and peptide motifs. Immunogenetics **50**(3-4), 213–219 (1999) 52, 53, 55, 56, 140

[112] Peters, B., Sette, A.: Generating quantitative models describing the sequence specificity of biological processes with the stabilized matrix method. BMC Bioinformatics **6**(1), 1 (2005) 52, 56, 57, 140

[113] Parker, K.C., Bednarek, M.A., Coligan, J.E.: Scheme for ranking potential HLA-A2 binding peptides based on independent binding of individual peptide side-chains. The Journal of Immunology **152**(1), 163–175 (1994) 52, 56, 57, 140

[114] Dönnes, P., Kohlbacher, O.: Integrated modeling of the major events in the MHC class I antigen processing pathway. Protein Science **14**(8), 2132–2140 (2005) 52, 53, 55, 56, 140

[115] Tenzer, S., Peters, B., Bulik, S., Schoor, O., Lemmel, C., Schatz, M., Kloetzel, P.-M., Rammensee, H.-G., Schild, H., Holzhütter, H.-G.: Modeling the MHC class I pathway by combining predictions of proteasomal cleavage, TAP transport and MHC class I binding. Cellular and Molecular Life Sciences CMLS **62**(9), 1025–1037 (2005) 52, 140

[116] Levy, A., Pitcovski, J., Frankenburg, S., Elias, O., Altuvia, Y., Margalit, H., Peretz, T., Golenser, J., Lotem, M.: A melanoma multiepitope polypeptide induces specific CD8+ T-cell response. Cellular immunology **250**(1), 24–30 (2007) 53, 55

[117] Aurisicchio, L., Fridman, A., Bagchi, A., Scarselli, E., La Monica, N., Ciliberto, G.: A novel minigene scaffold for therapeutic cancer vaccines. Oncoimmunology **3**(1), 27529 (2014) 53

[118] Depla, E., Van der Aa, A., Livingston, B.D., Crimi, C., Allosery, K., De Brabandere, V., Krakover, J., Murthy, S., Huang, M., Power, S., *et al.*: Rational design of a multiepitope vaccine encoding T-lymphocyte epitopes for treatment of chronic hepatitis B virus infections. Journal of virology **82**(1), 435–450 (2008) 55

[119] Bazhan, S., Karpenko, L., Ilyicheva, T., Belavin, P., Seregin, S., Danilyuk, N., Antonets, D., Ilyichev, A.: Rational design based synthetic polyepitope DNA vaccine for eliciting HIV-specific CD8+ T cell responses. Molecular immunology **47**(7), 1507–1515 (2010)

[120] Moss, S.F., Moise, L., Lee, D.S., Kim, W., Zhang, S., Lee, J., Rogers, A.B., Martin, W., De Groot, A.S.: HelicoVax: epitope-based therapeutic Helicobacter pylori vaccination in a mouse model. Vaccine **29**(11), 2085–2091 (2011) 55

[121] Calis, J.J., Reinink, P., Keller, C., Kloetzel, P.M., Keşmir, C.: Role of peptide processing predictions in T cell epitope identification: contribution of different prediction programs. Immunogenetics **67**(2), 85–93 (2015) 58, 100

[122] Ryu, J.K., Kim, H.S., Nam, D.H.: Current status and perspectives of biopharmaceutical drugs. Biotechnology and Bioprocess Engineering **17**(5), 900–911 (2012) 59

[123] Walsh, G.: Biopharmaceutical benchmarks 2014. Nature biotechnology **32**(10), 992–1000 (2014) 59

[124] Krieckaert, C., Rispens, T., Wolbink, G.: Immunogenicity of biological therapeutics: from assay to patient. Current opinion in rheumatology **24**(3), 306–311 (2012) 59

[125] Perini, P., Facchinetti, A., Bulian, P., Massaro, A.R., Pascalis, D.D., Bertolotto, A., Biasi, G., Gallo, P.: Interferon-beta (INF-beta) antibodies in interferon-beta1a- and interferon-beta1b-treated multiple sclerosis patients. Prevalence, kinetics, cross-reactivity, and factors enhancing interferon-beta immunogenicity in vivo. Eur Cytokine Netw **12**(1), 56–61 (2001) 61

[126] Singh, S.K.: Impact of product-related factors on immunogenicity of biotherapeutics. Journal of pharmaceutical sciences **100**(2), 354–387 (2011) 61

[127] Ross, C., Clemmesen, K.M., Svenson, M., Soelberg Sørensen, P., Koch-Henriksen, N., Lange Skovgaard, G., Bendtzen, K.: Immunogenicity of interferon-β in multiple sclerosis patients: Influence of preparation, dosage, dose frequency, and route of administration. Annals of neurology **48**(5), 706–712 (2000) 61

[128] Rosenschein, U., Lenz, R., Radnay, J., Ben, T.T., Rozenszajn, L.: Streptokinase immunogenicity in thrombolytic therapy for acute myocardial infarction. Israel journal of medical sciences **27**(10), 541–545 (1991) 61

[129] Grauer, A., Frank-Raue, K., Schroth, J., Raue, F., Ziegler, R.: Neutralizing antibodies against salmon calcitonin. The cause of a treatment failure in Paget's disease. Deutsche medizinische Wochenschrift (1946) **119**(14), 507–510 (1994) 61

[130] Chaffee, S., Mary, A., Stiehm, E., Girault, D., Fischer, A., Hershfield, M.S.: IgG antibody response to polyethylene glycol-modified adenosine deaminase in patients with adenosine deaminase deficiency. Journal of Clinical Investigation **89**(5), 1643 (1992) 61

[131] Oberg, K., Alm, G., Magnusson, A., Lundqvist, G., Theodorsson, E., Wide, L., Wilander, E.: Treatment of malignant carcinoid tumors with recombinant interferon alfa-2b: development of neutralizing interferon antibodies and possible loss of antitumor activity. Journal of the National Cancer Institute **81**(7), 531–535 (1989) 61

[132] Prabhakar, S., Muhlfelder, T.: Antibodies to recombinant human erythropoietin causing pure red cell aplasia. Clinical nephrology **47**(5), 331–335 (1997)

[133] Kontsek, P., Liptakova, H., Kontsekova, E.: Immunogenicity of interferon-alpha 2 in therapy: structural and physiological aspects. Acta virologica **43**(1), 63–70 (1999)

[134] Zang, Y., Yang, D., Hong, J., Tejada-Simon, M., Rivera, V., Zhang, J.: Immunoregulation and blocking antibodies induced by interferon beta treatment in MS. Neurology **55**(3), 397–404 (2000) 61

[135] Prescott, R., Nakai, H., Saenko, E.L., Scharrer, I., Nilsson, I.M., Humphries, J.E., Hurst, D., Bray, G., Scandella, D., *et al.*: The inhibitor antibody response is more complex in hemophilia A patients than in most nonhemophiliacs with factor VIII autoantibodies. Blood **89**(10), 3663–3671 (1997) 61

[136] Schernthaner, G., Borkenstein, M., Fink, M., Mayr, W., Menzel, J., Schober, E.: Immunogenicity of human insulin (Novo) or pork monocomponent insulin in HLA-DR-typed insulin-dependent diabetic individuals. Diabetes Care **6**, 43–48 (1982) 61

[137] Link, J., Ryner, M.L., Fink, K., Hermanrud, C., Lima, I., Brynedal, B., Kockum, I., Hillert, J., Fogdell-Hahn, A.: Human leukocyte antigen genes and interferon beta preparations influence risk of developing neutralizing anti-drug antibodies in multiple sclerosis. PloS one **9**(3), 90479 (2014) 61

[138] Gribben, J., Devereux, S., Thomas, N., Keim, M., Jones, H., Goldstone, A., Linch, D.: Development of antibodies to unprotected glycosylation sites on recombinant human GM-CSF. The Lancet **335**(8687), 434–437 (1990) 61

[139] Macdougall, I.C.: Novel erythropoiesis stimulating protein. In: Seminars in Nephrology. 4, vol. 20, pp. 375–381 (2000) 61

[140] Sinclair, A.M., Elliott, S.: Glycoengineering: the effect of glycosylation on the properties of therapeutic proteins. Journal of pharmaceutical sciences **94**(8), 1626–1635 (2005) 61

[141] Bailon, P., Won, C.-Y.: PEG-modified biopharmaceuticals. Expert opinion on drug delivery **6**(1), 1–16 (2009) 61

[142] Nelson, A.L., Dhimolea, E., Reichert, J.M.: Development trends for human monoclonal antibody therapeutics. Nature reviews Drug discovery **9**(10), 767–774 (2010) 62

[143] Hwang, W.Y.K., Foote, J.: Immunogenicity of engineered antibodies. Methods **36**(1), 3–10 (2005) 62

[144] Harding, F.A., Stickler, M.M., Razo, J., DuBridge, R.: The immunogenicity of humanized and fully human antibodies: residual immunogenicity resides in the CDR regions. In: MAbs. 3, vol. 2, pp. 256–265 (2010). Taylor & Francis 62

[145] Baker, M.P., Jones, T.D.: Identification and removal of immunogenicity in therapeutic proteins. Current opinion in drug discovery & development **10**(2), 219–227 (2007) 62

[146] Mazor, R., Vassall, A.N., Eberle, J.A., Beers, R., Weldon, J.E., Venzon, D.J., Tsang, K.Y., Benhar, I., Pastan, I.: Identification and elimination of an immunodominant T-cell epitope in recombinant immunotoxins based on Pseudomonas exotoxin A. Proceedings of the National Academy of Sciences **109**(51), 3597–3603 (2012) 62

[147] Onda, M., Beers, R., Xiang, L., Nagata, S., Wang, Q.-c., Pastan, I.: An immunotoxin with greatly reduced immunogenicity by identification and removal of B cell epitopes. Proceedings of the National Academy of Sciences **105**(32), 11311–11316 (2008)

[148] Tangri, S., Mothé, B.R., Eisenbraun, J., Sidney, J., Southwood, S., Briggs, K., Zinckgraf, J., Bilsel, P., Newman, M., Chesnut, R., *et al.*: Rationally engineered therapeutic proteins with reduced immunogenicity. The Journal of Immunology **174**(6), 3187–3196 (2005) 62

[149] De Groot, A., Knopp, P., Martin, W.: De-immunization of therapeutic proteins by T-cell epitope modification. Developments in biologicals **122**, 171–194 (2004) 62

[150] Moise, L., Song, C., Martin, W.D., Tassone, R., De Groot, A.S., Scott, D.W.: Effect of HLA DR epitope de-immunization of Factor VIII in vitro and in vivo. Clinical Immunology **142**(3), 320–331 (2012) 62, 83, 84

[151] Cantor, J.R., Yoo, T.H., Dixit, A., Iverson, B.L., Forsthuber, T.G., Georgiou, G.: Therapeutic enzyme deimmunization by combinatorial T-cell epitope removal using neutral drift. Proceedings of the National Academy of Sciences **108**(4), 1272–1277 (2011) 62

[152] Parker, A.S., Choi, Y., Griswold, K.E., Bailey-Kellogg, C.: Structure-guided deimmunization of therapeutic proteins. Journal of Computational Biology **20**(2), 152–165 (2013) 63, 83

[153] Parker, A.S., Griswold, K.E., Bailey-Kellogg, C.: Optimization of therapeutic proteins to delete T-cell epitopes while maintaining beneficial residue interactions. Journal of bioinformatics and computational biology **9**(02), 207–229 (2011) 63

[154] Marks, D.S., Colwell, L.J., Sheridan, R., Hopf, T.A., Pagnani, A., Zecchina, R., Sander, C.: Protein 3D structure computed from evolutionary sequence variation. PloS one **6**(12), 28766 (2011) 63

[155] Morcos, F., Pagnani, A., Lunt, B., Bertolino, A., Marks, D.S., Sander, C., Zecchina, R., Onuchic, J.N., Hwa, T., Weigt, M.: Direct-coupling analysis of residue coevolution captures native contacts across many protein families. Proceedings of the National Academy of Sciences **108**(49), 1293–1301 (2011) 63

[156] Marks, D.S., Hopf, T.A., Sander, C.: Protein structure prediction from sequence variation. Nature biotechnology **30**(11), 1072–1080 (2012) 64

[157] Hopf, T.A., Schärfe, C.P., Rodrigues, J.P., Green, A.G., Kohlbacher, O., Sander, C., Bonvin, A.M., Marks, D.S.: Sequence co-evolution gives 3D contacts and structures of protein complexes. Elife **3**, 03430 (2014) 84

[158] Hopf, T.A., Morinaga, S., Ihara, S., Touhara, K., Marks, D.S., Benton, R.: Amino acid coevolution reveals three-dimensional structure and functional domains of insect odorant receptors. Nature communications **6** (2015) 63, 64, 71

[159] Shekhar, K., Ruberman, C.F., Ferguson, A.L., Barton, J.P., Kardar, M., Chakraborty, A.K.: Spin models inferred from patient-derived viral sequence data faithfully describe HIV fitness landscapes. Physical review E **88**(6), 062705 (2013) 64, 65

[160] Hopf, T.A., Ingraham, J.B., Poelwijk, F.J., Springer, M., Sander, C., Marks, D.S.: Quantification of the effect of mutations using a global probability model of natural sequence variation. arXiv preprint arXiv:1510.04612 (2015) 66, 67

[161] Figliuzzi, M., Jacquier, H., Schug, A., Tenaillon, O., Weigt, M.: Coevolutionary landscape inference and the context-dependence of mutations in beta-lactamase TEM-1. Molecular biology and evolution **33**(1), 268–280 (2016) 64, 65

[162] Ekeberg, M., Lövkvist, C., Lan, Y., Weigt, M., Aurell, E.: Improved contact prediction in proteins: using pseudolikelihoods to infer Potts models. Physical Review E **87**(1), 012707 (2013) 64, 65

[163] Kamisetty, H., Ovchinnikov, S., Baker, D.: Assessing the utility of coevolution-based residue–residue contact predictions in a sequence-and structure-rich era. Proceedings of the National Academy of Sciences **110**(39), 15674–15679 (2013) 64

[164] Dunn, S.D., Wahl, L.M., Gloor, G.B.: Mutual information without the influence of phylogeny or entropy dramatically improves residue contact prediction. Bioinformatics **24**(3), 333–340 (2008) 65

[165] Lapedes, A., Giraud, B., Jarzynski, C.: Using sequence alignments to predict protein structure and stability with high accuracy. arXiv preprint arXiv:1207.2484 (2012) 65

[166] Kingsford, C.L., Chazelle, B., Singh, M.: Solving and analyzing side-chain positioning problems using linear and integer programming. Bioinformatics **21**(7), 1028–1039 (2005) 66, 67

[167] Zhang, L., Chen, Y., Wong, H.-S., Zhou, S., Mamitsuka, H., Zhu, S.: TEPITOPEpan: extending TEPITOPE for peptide binding prediction covering over 700 HLA-DR molecules. PLoS One **7**(2), 30483 (2012) 66, 140

[168] Sturniolo, T., Bono, E., Ding, J., Raddrizzani, L., Tuereci, O., Sahin, U., Braxenthaler, M., Gallazzi, F., Protti, M.P., Sinigaglia, F., *et al.*: Generation of tissue-specific and promiscuous

HLA ligand databases using DNA microarrays and virtual HLA class II matrices. Nature biotechnology **17**(6), 555–561 (1999) 66

[169] Poladian, L., Jermiin, L.: Multi-objective evolutionary algorithms and phylogenetic inference with multiple data sets. Soft Computing **10**(4), 359–368 (2006) 67

[170] van Someren, E.P., Wessels, L.F., Backer, E., Reinders, M.J.: Multi-criterion optimization for genetic network modeling. Signal Processing **83**(4), 763–775 (2003) 67

[171] Koduru, P., Das, S., Welch, S., Roe, J.L.: A multi-objective GA-simplex hybrid approach for gene regulatory network models. In: Evolutionary Computation, 2004. CEC2004. Congress On, vol. 2, pp. 2084–2091 (2004). IEEE

[172] Schwarz, R., Musch, P., von Kamp, A., Engels, B., Schirmer, H., Schuster, S., Dandekar, T.: YANA–a software tool for analyzing flux modes, gene-expression and enzyme activities. Bmc Bioinformatics **6**(1), 135 (2005) 67

[173] Day, R.O., Zydallis, J.B., Lamont, G.B., Pachter, R.: Solving the protein structure prediction problem through a multiobjective genetic algorithm. Nanotechnology **2**, 32–35 (2002) 67

[174] Schulze-Kremer, S.: Application of evolutionary computation to protein folding with specialized operators. Evolutionary Computation in Bioinformatics, 163–192 (2003)

[175] Cutello, V., Narzisi, G., Nicosia, G.: A multi-objective evolutionary approach to the protein structure prediction problem. Journal of The Royal Society Interface **3**(6), 139–151 (2006) 67

[176] Putz, H., Schön, J., Jansen, M.: Combined method for ab initio structure solution from powder diffraction data. Journal of applied crystallography **32**(5), 864–870 (1999) 67

[177] Lanning, O.J., Habershon, S., Harris, K.D., Johnston, R.L., Kariuki, B.M., Tedesco, E., Turner, G.W.: Definition of aguiding function'in global optimization: a hybrid approach combining energy and R-factor in structure solution from powder diffraction data. Chemical Physics Letters **317**(3), 296–303 (2000) 67

[178] Roytberg, M., Semionenkov, M., Tabolina, O.Y.: Pareto-optimal alignment of biological sequences. Biofizika **44**(4), 565–577 (1999) 67

[179] Zwir, I., Zaliz, R.R., Ruspini, E.H.: Automated biological sequence description by genetic multiobjective generalized clustering. Annals of the New York Academy of Sciences **980**(1), 65–82 (2002) 67

[180] Cottrell, S.J., Gillet, V.J., Taylor, R., Wilton, D.J.: Generation of multiple pharmacophore hypotheses using multiobjective optimisation techniques. Journal of computer-aided molecular design **18**(11), 665–682 (2004) 67

[181] Baum, D.: Multiple semi-flexible 3D superposition of drug-sized molecules. Springer, Berlin (2005) 67

[182] Zitzler, E., Laumanns, M., Thiele, L.: SPEA2: Improving the strength Pareto evolutionary algorithm. Eidgenössische Technische Hochschule Zürich (ETH), Institut für Technische Informatik und Kommunikationsnetze (TIK) Zürich, Switzerland, Heidelberg (2001) 68

[183] Deb, K., Pratap, A., Agarwal, S., Meyarivan, T.: A fast and elitist multiobjective genetic algorithm: NSGA-II. Evolutionary Computation, IEEE Transactions on **6**(2), 182–197 (2002)

[184] Knowles, J., Corne, D.: The pareto archived evolution strategy: A new baseline algorithm for pareto multiobjective optimisation. In: Evolutionary Computation, 1999. CEC 99. Proceedings of the 1999 Congress On, vol. 1 (1999). IEEE 68

[185] Sierra, M.R., Coello, C.A.C.: Improving PSO-based multi-objective optimization using crowding, mutation and -dominance. In: Evolutionary Multi-criterion Optimization, pp. 505–519 (2005). Springer 68

[186] Koduru, P., Das, S., Welch, S.M.: Multi-objective hybrid PSO using μ-fuzzy dominance. In: Proceedings of the 9th Annual Conference on Genetic and Evolutionary Computation, pp. 853–860 (2007). ACM 68

[187] Nebro, A.J., Luna, F., Alba, E., Dorronsoro, B., Durillo, J.J., Beham, A.: AbYSS: Adapting scatter search to multiobjective optimization. Evolutionary Computation, IEEE Transactions on **12**(4), 439–457 (2008) 68

[188] Zitzler, E., Thiele, L., Laumanns, M., Fonseca, C.M., Da Fonseca, V.G.: Performance assessment of multiobjective optimizers: an analysis and review. Evolutionary Computation, IEEE Transactions on **7**(2), 117–132 (2003) 71

[189] Johnson, L.S., Eddy, S.R., Portugaly, E.: Hidden Markov model speed heuristic and iterative HMM search procedure. BMC bioinformatics **11**(1), 1 (2010) 71

[190] Schymkowitz, J., Borg, J., Stricher, F., Nys, R., Rousseau, F., Serrano, L.: The FoldX web server: an online force field. Nucleic acids research **33**(suppl 2), 382–388 (2005) 71

[191] Orlova, N., Kovnir, S., Vorobiev, I., Gabibov, A., Vorobiev, A.: Blood clotting factor VIII: from evolution to therapy. Acta Naturae **5**(2 (17)) (2013) 75

[192] Shen, B.W., Spiegel, P.C., Chang, C.-H., Huh, J.-W., Lee, J.-S., Kim, J., Kim, Y.-H., Stoddard, B.L.: The tertiary structure and domain organization of coagulation factor VIII. Blood **111**(3), 1240–1247 (2008) 75

[193] Bhopale, G., Nanda, R.: Blood coagulation factor VIII: An overview. Journal of biosciences **28**(6), 783–789 (2003) 75

[194] Pallister, C., Watson, M.: Haematology, 2nd edn. Scion, Banbury (2010) 75

[195] Peyvandi, F., Garagiola, I., Young, G.: The past and future of haemophilia: diagnosis, treatments, and its complications. The Lancet (2016) 75

[196] Antonarakis, S.E., Rossiter, J., Young, M., Horst, J., De Moerloose, P., Sommer, S., Ketterling, R.P., Kazazian, H.J., Negrier, C., Vinciguerra, C.: Factor VIII gene inversions in severe hemophilia A: results of an international consortium study. Blood **86**(6), 2206–2212 (1995) 75

[197] Mannucci, P.M., Tuddenham, E.G.: The hemophilias - from royal genes to gene therapy. New England Journal of Medicine **344**(23), 1773–1779 (2001) 76

[198] Valentino, L.: Blood-induced joint disease: the pathophysiology of hemophilic arthropathy. Journal of Thrombosis and Haemostasis **8**(9), 1895–1902 (2010) 76

[199] Collins, P.W.: Personalized prophylaxis. Haemophilia **18**(s4), 131–135 (2012) 76

[200] Astermark, J.: Inhibitor development: patient-determined risk factors. Haemophilia **16**(102), 66–70 (2010) 76

[201] Lacroix-Desmazes, S., Misra, N., Bayry, J., Artaud, C., Drayton, B., Kaveri, S., Kazatchkine, M.: Pathophysiology of inhibitors to factor VIII in patients with haemophilia A. Haemophilia **8**(3), 273–279 (2002) 76

[202] Astermark, J., Oldenburg, J., Escobar, M., White, G.C., Berntorp, E., study group, M.I.B.S., *et al.*: The Malmo International Brother Study (MIBS). Genetic defects and inhibitor development in siblings with severe hemophilia A. Haematologica **90**(7), 924–931 (2005) 76

[203] Lavigne-Lissalde, G., Schved, J.-F., Granier, C., Villard, S., *et al.*: Anti-factor VIII antibodies: a 2005 update. Thrombosis and haemostasis **94**(4), 760–769 (2005) 76

[204] Meeks, S.L., Cox, C.L., Healey, J.F., Parker, E.T., Doshi, B.S., Gangadharan, B., Barrow, R.T., Lollar, P.: A major determinant of the immunogenicity of factor VIII in a murine model is independent of its procoagulant function. Blood **120**(12), 2512–2520 (2012) 76

[205] Walter, J.D., Werther, R.A., Brison, C.M., Cragerud, R.K., Healey, J.F., Meeks, S.L., Lollar, P., Spiegel, P.C.: Structure of the factor VIII C2 domain in a ternary complex with 2 inhibitor antibodies reveals classical and nonclassical epitopes. Blood **122**(26), 4270–4278 (2013) 76, 83

[206] Liu, Z., Lin, L., Yuan, C., Nicolaes, G.A., Chen, L., Meehan, E.J., Furie, B., Furie, B., Huang, M.: Trp2313-His2315 of Factor VIII C2 Domain is Involved in Membrane Binding. Structure of a Complex Between the C2 Domainand an Inhibitor of Membrane Binding. Journal of Biological Chemistry **285**(12), 8824–8829 (2010) 76, 81

[207] Feldhahn, M., Dönnes, P., Thiel, P., Kohlbacher, O.: FRED - framework for T-cell epitope detection. Bioinformatics **25**(20), 2758–2759 (2009) 85, 86, 140

[208] Farrell, D., Gordon, S.V.: Epitopemap: a web application for integrated whole proteome epitope prediction. BMC bioinformatics **16**(1), 221 (2015) 85

[209] Vita, R., Overton, J.A., Greenbaum, J.A., Ponomarenko, J., Clark, J.D., Cantrell, J.R., Wheeler, D.K., Gabbard, J.L., Hix, D., Sette, A., *et al.*: The immune epitope database (IEDB) 3.0. Nucleic acids research **43**(D1), 405–412 (2015) 86, 93

[210] Feldhahn, M., Thiel, P., Schuler, M.M., Hillen, N., Stevanović, S., Rammensee, H.-G., Kohlbacher, O.: EpiToolKit - a web server for computational immunomics. Nucleic acids research **36**(suppl 2), 519–522 (2008) 86

[211] Wang, K., Li, M., Hakonarson, H.: ANNOVAR: functional annotation of genetic variants from high-throughput sequencing data. Nucleic acids research **38**(16), 164–164 (2010) 87, 91, 96

[212] McLaren, W., Gil, L., Hunt, S.E., Riat, H.S., Ritchie, G.R., Thormann, A., Flicek, P., Cunningham, F.: The Ensembl Variant Effect Predictor. Genome biology **17**(1), 1 (2016) 87

[213] Consortium, U., *et al.*: The universal protein resource (UniProt). Nucleic acids research **36**(suppl 1), 190–195 (2008) 87, 91

[214] Pruitt, K.D., Tatusova, T., Maglott, D.R.: NCBI reference sequences (RefSeq): a curated non-redundant sequence database of genomes, transcripts and proteins. Nucleic acids research **35**(suppl 1), 61–65 (2007) 87, 91

[215] Yates, A., Akanni, W., Amode, M.R., Barrell, D., Billis, K., Carvalho-Silva, D., Cummins, C., Clapham, P., Fitzgerald, S., Gil, L., *et al.*: Ensembl 2016. Nucleic acids research **44**(D1), 710–716 (2016) 87

[216] Smedley, D., Haider, S., Ballester, B., Holland, R., London, D., Thorisson, G., Kasprzyk, A.: BioMart–biological queries made easy. BMC genomics **10**(1), 1 (2009) 87

[217] Python Consortium: PEP 3119 - Introducing Abstract Base Classes. `https://www.python.org/dev/peps/pep-3119` 87

[218] Feldhahn, M., Dönnes, P., Schubert, B., Schilbach, K., Rammensee, H.-G., Kohlbacher, O.: miHA-Match: computational detection of tissue-specific minor histocompatibility antigens. Journal of immunological methods **386**(1), 94–100 (2012) 87

[219] Appelbaum, F.R.: The current status of hematopoietic cell transplantation. Annual review of medicine **54**(1), 491–512 (2003) 89

[220] Feng, X., Hui, K.M., Younes, H.M., Brickner, A.G.: Targeting minor histocompatibility antigens in graft versus tumor or graft versus leukemia responses. Trends in immunology **29**(12), 624–632 (2008) 89

[221] Sherry, S.T., Ward, M.-H., Kholodov, M., Baker, J., Phan, L., Smigielski, E.M., Sirotkin, K.: dbSNP: the NCBI database of genetic variation. Nucleic acids research **29**(1), 308–311 (2001) 91

[222] Schuler, M.M., Dönnes, P., Nastke, M.-D., Kohlbacher, O., Rammensee, H.-G., Stevanovic, S.: SNEP: SNP-derived epitope prediction program for minor H antigens. Immunogenetics **57**(11), 816–820 (2005) 91

[223] Toussaint, N.C., Kohlbacher, O.: OptiTope - a web server for the selection of an optimal set of peptides for epitope-based vaccines. Nucleic acids research **37**(suppl 2), 617–622 (2009) 91, 140

[224] Squires, R.B., Noronha, J., Hunt, V., García-Sastre, A., Macken, C., Baumgarth, N., Suarez, D., Pickett, B.E., Zhang, Y., Larsen, C.N., *et al.*: Influenza research database: an integrated bioinformatics resource for influenza research and surveillance. Influenza and other respiratory viruses **6**(6), 404–416 (2012) 93

[225] Lundegaard, C., Lamberth, K., Harndahl, M., Buus, S., Lund, O., Nielsen, M.: NetMHC-3.0: accurate web accessible predictions of human, mouse and monkey MHC class I affinities for peptides of length 8–11. Nucleic acids research **36**(suppl 2), 509–512 (2008) 93, 140

[226] Network, C.G.A., *et al.*: Comprehensive molecular characterization of human colon and rectal cancer. Nature **487**(7407), 330–337 (2012) 96

[227] Cerami, E., Gao, J., Dogrusoz, U., Gross, B.E., Sumer, S.O., Aksoy, B.A., Jacobsen, A., Byrne, C.J., Heuer, M.L., Larsson, E., *et al.*: The cBio cancer genomics portal: an open platform for exploring multidimensional cancer genomics data. Cancer discovery **2**(5), 401–404 (2012) 97

[228] Moreews, F., Sallou, O., Ménager, H., Monjeaud, C., Blanchet, C., Collin, O., et al.: BioShaDock: a community driven bioinformatics shared Docker-based tools registry. F1000Research **4** (2015) 98

[229] Liu, X., Zhang, X., Caetano, T.: Bayesian models for structured sparse estimation via set cover prior. In: Joint European Conference on Machine Learning and Knowledge Discovery in Databases, Heidelberg, pp. 273–289 (2014). Springer 100

[230] Kowalewski, D.J., Stevanović, S.: Biochemical large-scale identification of MHC class I ligands. Antigen Processing: Methods and Protocols, 145–157 (2013) 100

[231] Vansteenwegen, P., Souffriau, W., Van Oudheusden, D.: The orienteering problem: A survey. European Journal of Operational Research **209**(1), 1–10 (2011) 100

[232] Myers, E.W.: Toward simplifying and accurately formulating fragment assembly. Journal of Computational Biology **2**(2), 275–290 (1995) 101

[233] De Groot, A.S., Marcon, L., Bishop, E.A., Rivera, D., Kutzler, M., Weiner, D.B., Martin, W.: HIV vaccine development by computer assisted design: the GAIA vaccine. Vaccine **23**(17), 2136–2148 (2005) 101

[234] Fischer, W., Perkins, S., Theiler, J., Bhattacharya, T., Yusim, K., Funkhouser, R., Kuiken, C., Haynes, B., Letvin, N.L., Walker, B.D., *et al.*: Polyvalent vaccines for optimal coverage of potential T-cell epitopes in global HIV-1 variants. Nature medicine **13**(1), 100–106 (2007)

[235] Barouch, D.H., Stephenson, K.E., Borducchi, E.N., Smith, K., Stanley, K., McNally, A.G., Liu, J., Abbink, P., Maxfield, L.F., Seaman, M.S., *et al.*: Protective efficacy of a global HIV-1 mosaic vaccine against heterologous SHIV challenges in rhesus monkeys. Cell **155**(3), 531–539 (2013) 101

[236] Pierce, N.A., Winfree, E.: Protein design is NP-hard. Protein Engineering **15**(10), 779–782 (2002) 102

[237] Chazelle, B., Kingsford, C., Singh, M.: A semidefinite programming approach to side chain positioning with new rounding strategies. INFORMS Journal on Computing **16**(4), 380–392 (2004) 102

[238] Yanover, C., Meltzer, T., Weiss, Y.: Linear programming relaxations and belief propagation–an empirical study. The Journal of Machine Learning Research **7**, 1887–1907 (2006) 102

[239] Sontag, D., Meltzer, T., Globerson, A., Jaakkola, T.S., Weiss, Y.: Tightening LP relaxations for MAP using message passing. arXiv preprint arXiv:1206.3288 (2012) 102

[240] Sturm, M., Bertsch, A., Gröpl, C., Hildebrandt, A., Hussong, R., Lange, E., Pfeifer, N., Schulz-Trieglaff, O., Zerck, A., Reinert, K., *et al.*: OpenMS–an open-source software framework for mass spectrometry. BMC bioinformatics **9**(1), 163 (2008) 103

[241] Dönnes, P., Elofsson, A.: Prediction of MHC class I binding peptides, using SVMHC. BMC bioinformatics **3**(1), 1 (2002) 140

[242] Bui, H.-H., Sidney, J., Peters, B., Sathiamurthy, M., Sinichi, A., Purton, K.-A., Mothé, B.R., Chisari, F.V., Watkins, D.I., Sette, A.: Automated generation and evaluation of specific MHC binding predictive tools: ARB matrix applications. Immunogenetics **57**(5), 304–314 (2005) 140

[243] Kim, Y., Sidney, J., Pinilla, C., Sette, A., Peters, B.: Derivation of an amino acid similarity matrix for peptide: MHC binding and its application as a Bayesian prior. BMC bioinformatics **10**(1), 1 (2009) 140

[244] Sidney, J., Assarsson, E., Moore, C., Ngo, S., Pinilla, C., Sette, A., Peters, B.: Quantitative peptide binding motifs for 19 human and mouse MHC class I molecules derived using positional scanning combinatorial peptide libraries. Immunome research **4**(1), 1 (2008) 140

[245] Zhang, H., Lund, O., Nielsen, M.: The PickPocket method for predicting binding specificities for receptors based on receptor pocket similarities: application to MHC-peptide binding. Bioinformatics **25**(10), 1293–1299 (2009) 140

[246] Hoof, I., Peters, B., Sidney, J., Pedersen, L.E., Sette, A., Lund, O., Buus, S., Nielsen, M.: NetMHCpan, a method for MHC class I binding prediction beyond humans. Immunogenetics **61**(1), 1–13 (2009) 140

[247] Sturniolo, T., Bono, E., Ding, J., Raddrizzani, L., Tuereci, O., Sahin, U., Braxenthaler, M., Gallazzi, F., Protti, M.P., Sinigaglia, F., *et al.*: Generation of tissue-specific and promiscuous

HLA ligand databases using DNA microarrays and virtual HLA class II matrices. Nature biotechnology **17**(6), 555–561 (1999) 140

[248] Nielsen, M., Lundegaard, C., Lund, O.: Prediction of MHC class II binding affinity using SMM-align, a novel stabilization matrix alignment method. BMC bioinformatics **8**(1), 1 (2007) 140

[249] Karosiene, E., Rasmussen, M., Blicher, T., Lund, O., Buus, S., Nielsen, M.: NetMHCIIpan-3.0, a common pan-specific MHC class II prediction method including all three human MHC class II isotypes, HLA-DR, HLA-DP and HLA-DQ. Immunogenetics **65**(10), 711–724 (2013) 140

[250] Toussaint, N.C., Feldhahn, M., Ziehm, M., Stevanović, S., Kohlbacher, O.: T-cell epitope prediction based on self-tolerance. In: Proceedings of the 2nd ACM Conference on Bioinformatics, Computational Biology and Biomedicine, pp. 584–588 (2011). ACM 140

[251] Stranzl, T., Larsen, M.V., Lundegaard, C., Nielsen, M.: NetCTLpan: pan-specific MHC class I pathway epitope predictions. Immunogenetics **62**(6), 357–368 (2010) 140

[252] Nielsen, M., Lundegaard, C., Lund, O., Keşmir, C.: The role of the proteasome in generating cytotoxic T-cell epitopes: insights obtained from improved predictions of proteasomal cleavage. Immunogenetics **57**(1-2), 33–41 (2005) 140

[253] Ginodi, I., Vider-Shalit, T., Tsaban, L., Louzoun, Y.: Precise score for the prediction of peptides cleaved by the proteasome. Bioinformatics **24**(4), 477–483 (2008) 140

[254] Peters, B., Bulik, S., Tampe, R., Van Endert, P.M., Holzhütter, H.-G.: Identifying MHC class I epitopes by predicting the TAP transport efficiency of epitope precursors. The Journal of Immunology **171**(4), 1741–1749 (2003) 140

[255] Doytchinova, I., Hemsley, S., Flower, D.R.: Transporter associated with antigen processing preselection of peptides binding to the MHC: a bioinformatic evaluation. The Journal of Immunology **173**(11), 6813–6819 (2004) 140

[256] Schubert, B., Kohlbacher, O.: Designing string-of-beads vaccines with optimal spacers. Genome medicine **8**(1), 1 (2016) 140

Appendix A

Abbreviations

Amino acids in a general are abbreviated in standard one-letter code. Mutations are denoted with the 1-one letter amino acid reference followed by the position of the mutation and the mutated amino acid as suffix.

AA	*Amino acid*
ADA	*Anti-drug antibody*
ANN	*Artificial neural network N*
APC	*Antigen presenting cell*
BCR	*B-Cell receptor*
BOMIP	*Bi-objective mixed integer program*
bp	*base pair*
B&B	*Branch-and-Bound*
B&C	*Branch-and-Cut*
CD	*Cluster of differentiation*
CI	*Confidence interval*
CDS	*Coding sequence*
CTD	*Common tool descriptor*
CTL	*Cytotoxic T lymphocyte*
EV	*Epitope-based vaccine*
ER	*Endoplasmic reticulum*
ERAAP	*Aminopeptidase associated with antigen processing*
ETK	*EpiToolKit*
Fab	*Antigen binding fragment*
Fc	*Fragment of crystallization*
FRED	*Framework for epitope detection*
GC	*Germinal center*
GKN	*Generic KNIME node*

GUB	*Global upper bound*
HLA	*Human leukocyte antigen*
HBV	*Hepatitis B virus*
IC_{50}	*Half maximal inhibitory concentration*
Ig	*Immunoglobulin*
Ii	*Invariant chain*
IL	*Interleukin*
ILP	*Integer linear program*
IDE	*Integrated development environment*
KNIME	*Konstanz information miner*
LKH	*Lin-Kernighan-Helsgaun heuristic*
LLB	*Local lower bound*
LP	*Linear program*
LO	*Lexicographic optimization*
mAb	*Monoclonal antibody*
MHC	*Major histocompatibility complex*
MILP	*Mixed integer linear program*
ML	*Machine learning*
MO	*Multiobjective optimization*
MO(MI)LP	*Multiobjective (mixed integer) linear program*
MSA	*Multiple sequence alignment*
NGS	*Next-generation sequencing*
PAMP	*Pathogen-associated molecular pattern*
pHLA	*HLA-peptide complex*
PM	*Precision medicine*
PRR	*Pattern recognition receptor*
PSSM	*Position-specific scoring matrix*
SBV	*String-of-beads vaccine*
SLP	*Synthetic long peptides*
SVM	*Support vector machine*
TAP	*Transporter associated with antigen processing*
T_H	*T-helper cell*
T_{REG}	*regulatory T cell*
TSP	*Traveling Salesman problem*
$T_{1/2}$	*Half-life*
vWF	*Von Willebrand factor*
WES	*Whole exome sequencing*
WGS	*Whole genome sequencing*

Appendix B

Notations

u	*Scalar value*
$\mathbf{u}$	*Vector*
$\mathbf{U}$	*Matrix*
$\mathcal{X}$	*Set of solutions, search space*
$\mathcal{X}_E$	*Set of efficient solutions*
$\mathcal{Y}$	*Objective space*
$\mathcal{Y}_N$	*Set of non-dominated points, Pareto front*
$\mathbf{x}$	*Discreet variables of a (MO)MILP*
$\mathbf{A} \in \mathbb{R}^{m \times n}$	*Constraint matrix of a (MO)MILP*
$\mathbf{c} \in \mathbb{R}^n$	*Coefficient vector of a MILP*
$\mathbf{b} \in \mathbb{R}^m$	*Right hand side vector of a (MO)MILP*
s	*Slack variable of an LP*
$\mathbf{y}^I$	*Ideal Point*
$z()$	*Objective function*
$\mathbf{z}() := (z_1(), .., z_n())$	*Vector of objective functions*
z	*Objective value*
$\mathbf{z} := (z_1, ..z_n)$	*Vector of objective values*
$\mathbf{B}, \mathbf{N}$	*Basic and non-basic constraint matrix of a LP*
$\mathcal{B}, \mathcal{N}$	*Indices of basic and non-basic variables*
$\mathbf{x}_\mathcal{B}, \mathbf{x}_\mathcal{N}$	*Basic and non-basic variables of a LP*
$\delta(i,j) := 1$ if $i = j$ else 0	*Kronecker delta*
$\phi_c()$	*Linear cleavage site likelihood model*
$\phi_I()$	*Linear immunogenicity model*
H	*Set of HLA alleles if not other stated*
E	*Set of epitopes if not other stated*
$S[i]$	*Indicates the i-th character of sequence S*
$\mathbb{R}^n_+$	$\{\mathbf{y} \in \mathbb{R}^n \vert \mathbf{y} \geq \mathbf{0}\}$
Σ	*Alphabet of all natural occurring amino acids*

Appendix C

Contributions

All ideas, approaches and results presented in this work were developed and discussed with Prof. Dr. Oliver Kohlbacher (OK). The following co-workers besides myself (BS) also contributed to the different projects:

- H.P. Brachvogel (HPB)
- Dr. P. Dönnes (PD)
- Dr. M. Feldhahn (MF)
- L. de la Garza (LD)
- Dr. T. Hopf (TP)
- C. Jürges (CJ)
- C. Mohr (CM)
- Prof. Dr. D. Marks (DM)
- C. Schärfe (CS)
- A. Szolek (AS)
- Dr. M. Sturm (MS)
- M. Walzer (MW)

Chapter 4: NGS-based HLA Genotyping using Combinatorial Optimization

AS, BS, and CM designed and implemented the HLA typing pipeline. BS designed the mathematical model. AS designed the pre-processing and HLA-reconstruction procedure. AS, CM, and BS designed, performed, and evaluated the experiments. CM, MF, AS, BS, MS prepared the data. BS, CM, AS, and OK wrote the manuscript. All authors read and approved the manuscript. OK designed the study. Text and figures from this manuscript appeared in the chapter.

Chapter 5: Designing String-of-beads vaccines with optimal spacer

BS designed implemented and evaluated the mathematical framework. BS and OK designed the study and wrote the manuscript. Text and figures from this manuscript appeared in the chapter.

Chapter 6: De-immunization of Biotherapeutics

BS designed and implemented the mathematical model as well as the bi-objective integer solver. BS, CS, and PD designed the evaluation. BS and CS gathered and performed the evaluation. TH provided code to infer the statistical fitness model. BS, CS, PD, DM, OK wrote the manuscript. BS and OK designed the study. Text and figures from this manuscript appeared in the chapter.

Chapter 7: Translational Immunoinformatics

BS and MW designed and implemented FRED 2. HPB, AS, and CM contributed source code. BS, MW, and OK wrote the manuscript. BS and OK designed the project. Text and figures from this manuscript appeared in the chapter.

BS designed and implemented EpiToolKit 2. HPB and CJ helped implementing the web-service. BS and OK wrote the manuscript. BS and OK designed the project. Text and figures from this manuscript appeared in the chapter.

BS implemented ImmunoNodes and extended the Generic KNIME Node. LD implemented the continues integration and helped to extend the Generic KNIME Node. CM and MW contributed an additional node and a workflow example not mentioned in this work. BS, LD, and OK wrote the manuscript. Text and figures from this manuscript appeared in the chapter.

Appendix D

Publications

In preparation

Schubert, B., Schärfe, C.S., Dönnes, P., Hopf, T., Marks, D. and Kohlbacher, O. **De-immunization of Factor VIII using the evolutionary Hamiltonian.**, in preparation.

Schubert, B., Del la Garza, L., Mohr, C., Walzer, M., and Kohlbacher, O. **ImmunoNodes - Bringing Immunoinformatics to the World of Workflows.** *Genome Medicine*, submitted on 12/02/2016.

2016

Schubert, B., and Kohlbacher, O. (2016). **Designing string-of-beads vaccines with optimal spacers.** *Genome Medicine*, Special Issue: *Immunogenomics in health and disease*, **8**(1), 1.

Schubert, B., Walzer, M., Brachvogel, H. P., Szolek, A., Mohr, C., and Kohlbacher, O. (2016). **FRED 2 - An Immunoinformatics Framework for Python.** *Bioinformatics*, btw113.

2015

Schubert, B., Brachvogel, H., Jürges, C., and Kohlbacher, O. (2015). **EpiToolKit - A Web-based Workbench for Vaccine Design.** *Bioinformatics*, **31**(13), 2211-2213.

2014

Szolek, A.*, **Schubert, B.***, Mohr, C.*, Sturm, M., Feldhahn, M., and Kohlbacher, O. (2014). **OptiType: precision HLA typing from next-generation sequencing data.** *Bioinformatics* **30**(23), 3310-3316.
*) Joint first author

Olabarriaga, S. D., Benabdelkader, A., Caan, M. W., Jaghoori, M. M., Krüger, J., de la Garza, L., Mohr, C., Schubert B., Danezi, A., and Kiss, T. (2014). **WS-PGRADE/gUSE-Based Science Gateways in Teaching.** *In Science Gateways for Distributed Computing Infrastructures.* Springer International Publication.

2013

Schubert, B., Lund, O. and Nielsen, M. (2013). **Evaluation of peptide selection approaches for epitope-based vaccine design** *Tissue antigens* **82**, 243-251.

2012

Feldhahn, M., Dönnes, P., **Schubert, B.**, Schilbach, K., Rammensee, H.-G. and Kohlbacher, O. (2012). **miHA-Match: Computational detection of tissue-specific minor histocompatibility antigens.** *Journal of Immunological Methods* **386**(1), 94-100.

Appendix E

Supporting Figures

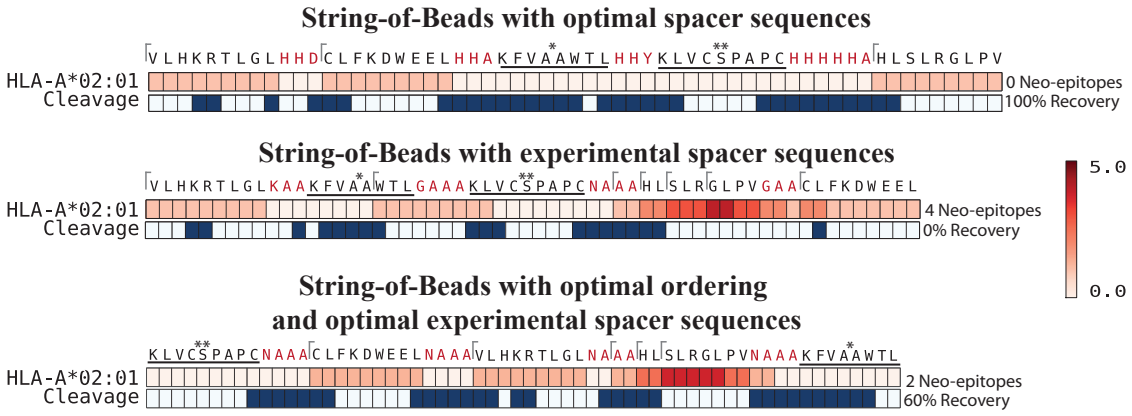


Figure E.1: Comparison of experimentally used and *in silico* designed spacers based on the polypeptide proposed by Ding *et al.* . Red bars represent predicted epitopes and the intensity indicates overlapping epitopes at that position. The blue rectangles represent predicted C-terminal cleavage sites. Spacer sequences are marked in red. A tick indicates the start position of a predicted nine-mer epitope. Epitope and cleavage site prediction were performed with SYFPEITHI and PCM, respectively. A nine-mer was predicted as an epitope if its predicted score was equal to or above a threshold of 20 (default threshold of SYFPEITHI). A cleavage site was said to be cleaved if the predicted PCM score was above zero. An epitope was defined as recovered if both the preceding and succeeding cleavage sites were predicted to be cleaved

Input Summary

Number of candidate epitopes: 1539

Number of target alleles: 47

Prediction Method: custom

Constraints

Maximum number of epitopes to select = 10

Epitope conservation ≥ 20.0%

Covered alleles ≥ 23

Covered antigens ≥ 5

Results

Selected epitopes: 10

Covered antigens: 5 of 10

Covered alleles: 26 of 47

Locus coverage:

A	84.00
B	66.48
C	93.75

Population converage: 99.66

Epitope	Conservation	Fraction of overall immunogenicity	Covered alleles	Covered antigens
ATYQRTRAL	95.9%	0.08	B*15:17 C*03:03 B*07:02 C*14:02 C*12:03 B*08:01 C*15:02 B*14:02 C*07:01	NP
FLARSALIL	99.8%	0.09	A*02:17 A*02:06 A*02:01 A*02:02 A*02:03 C*03:03 C*14:02 C*12:03 A*02:19 B*39:01 A*02:16 A*02:12	NP
FMQALQLLL	63.4%	0.09	B*15:01 A*02:17 A*29:02 A*02:06 A*02:01 A*02:02 A*02:03 C*14:02 A*02:19 B*39:01 A*02:16 A*02:12 B*08:03	NS2
FMYSDFHFI	99.7%	0.12	A*24:02 A*24:03 B*15:01 A*02:17 A*29:02 A*02:06 A*02:01 A*02:02 A*02:03 C*03:03 C*14:02 A*69:01 C*12:03 A*02:19 B*39:01 A*02:16 A*02:12	PA
FVANFSMEL	99.8%	0.11	A*02:17 A*02:06 A*02:01 A*02:02 A*02:03 C*03:03 C*14:02 B*35:01 A*69:01 C*12:03 A*02:19 A*02:16 A*02:12 C*15:02	PB1
FVRQCFNPM	99.2%	0.08	B*15:01 A*02:06 A*02:02 C*03:03 B*07:02 C*14:02 B*35:01 C*12:03 B*08:01	PA
LLIDGTASL	90.9%	0.09	B*15:01 A*02:17 A*02:06 A*02:01 A*02:02 A*02:03 C*03:03 C*14:02 A*69:01 C*12:03 A*02:19 A*02:16 A*02:12	PB1
MMMGMFNML	100.0%	0.12	A*24:02 A*24:03 B*15:01 A*02:17 A*29:02 A*02:06 A*02:01 A*02:02 A*02:03 C*03:03 C*14:02 A*69:01 C*12:03 A*02:19 B*39:01 A*02:16 A*02:12 B*08:01	PB1
WMMAMRYPI	56.8%	0.11	A*24:02 B*15:01 A*02:17 A*29:02 A*02:06 A*02:01 A*02:02 A*02:03 C*14:02 A*69:01 C*12:03 A*02:19 B*39:01 A*02:16 A*02:12 B*08:01 B*08:03	PB2
YLMAWKQVL	39.5%	0.12	B*15:01 A*02:17 A*02:06 A*02:01 A*02:02 A*02:03 C*03:03 C*14:02 A*69:01 C*12:03 A*02:19 B*39:01 A*02:16 A*02:12 B*08:01 C*15:02 C*07:01 C*07:02	PA

Download as CSV

Figure E.2: Epitope Selection for an influenza dataset consisting of H1N1 and H3N5 strains. The epitope set was optimized for the European population. NetMHC was used for epitope discover and default constraints of 50% HLA allele and antigen coverage and 20% epitope conservation was used.

Appendix F

Supporting Tables

CRC		1000 Genomes exome							
Sample ID	Run ID	Sample ID	Run ID	Sample ID	Run ID	Sample ID	Run ID	Sample ID	Run ID
17	SRR396926	NA06985	SRR709972	NA18537	ERR032033	NA18971	SRR078842	NA19207	SRR081256
	SRR396998	NA06994	SRR070528	NA18537	ERR032034	NA18972	SRR077490	NA19209	SRR077489
	SRR397070	NA06994	SRR070819	NA18542	ERR031855	NA18972	SRR081255	NA19209	SRR077859
	SRR397142	NA07000	SRR766039	NA18545	ERR031856	NA18973	SRR077861	NA19210	SRR078845
		NA07048	SRR099452	NA18547	ERR031957	NA18973	SRR078846	NA19210	SRR081222
20	SRR396928	NA07056	SRR764718	NA18550	ERR031958	NA18974	SRR077456	NA19222	SRR748214
	SRR397000	NA07357	SRR764689	NA18552	ERR031959	NA18974	SRR081248	NA19223	SRR071186
	SRR397072	NA07357	SRR764690	NA18555	ERR031857	NA18975	SRR078849	NA19223	SRR071193
	SRR397144	NA10847	SRR070531	NA18558	ERR031960	NA18975	SRR081225	NA19238	SRR071173
		NA10847	SRR070823	NA18561	ERR031858	NA18976	SRR077451	NA19238	SRR071195
42	SRR396942	NA10851	SRR766044	NA18562	ERR031859	NA18976	SRR077757	NA19238	SRR792121
	SRR397014	NA11829	SRR710128	NA18563	ERR031860	NA18978	SRR716650	NA19238	SRR792165
	SRR397086	NA11830	SRR766026	NA18564	ERR031861	NA18980	SRR716652	NA19239	SRR792097
	SRR397158	NA11831	SRR709975	NA18566	ERR031862	NA18980	SRR716653	NA19239	SRR792159
		NA11832	SRR766003	NA18570	ERR031863	NA18981	SRR077477	NA19240	SRR792091
49	SRR396946	NA11840	SRR070532	NA18571	ERR031868	NA18981	SRR077751	NA19240	SRR792767
	SRR397018	NA11840	SRR070809	NA18572	ERR031869	NA18987	SRR077491	NA20313	SRR359098
	SRR397090	NA11881	SRR766021	NA18573	ERR031870	NA18987	SRR077853	NA20313R	SRR359108
	SRR397162	NA11992	SRR701474	NA18576	ERR031871	NA18990	SRR077454	HG01756	SRR359102
		NA11994	SRR701475	NA18577	ERR032035	NA18990	SRR077486	HG01757	SRR359103
53	SRR396949	NA11995	SRR766010	NA18577	ERR032036	NA18991	SRR077450	HG01872	SRR359298
	SRR397021	NA12003	SRR766061	NA18579	ERR032037	NA18991	SRR077855	HG01873	SRR359295
	SRR397093	NA12004	SRR766059	NA18579	ERR032038	NA18992	SRR716428	HG01886	SRR360655
	SRR397165	NA12005	SRR718067	NA18582	ERR031961	NA18994	SRR716431	HG01953	SRR360288
		NA12006	SRR716422	NA18592	ERR031962	NA18995	SRR764775	HG01968	SRR360391
65	SRR396959	NA12043	SRR716423	NA18593	ERR034531	NA18997	SRR702078	HG02014	SRR360148
	SRR397031	NA12043	SRR716424	NA18603	ERR031872	NA18998	SRR766013	HG02057	SRR359301
	SRR397103	NA12044	SRR766060	NA18605	ERR031873	NA18999	SRR112297		
	SRR397175	NA12144	SRR766058	NA18608	ERR031874	NA19000	SRR099528		
		NA12154	SRR702067	NA18609	ERR031875	NA19003	SRR099532		
66	SRR397206	NA12155	SRR702068	NA18611	ERR031876	NA19005	SRR715906		
	SRR397266	NA12156	SRR764691	NA18612	ERR034593	NA19007	SRR099549		
	SRR397326	NA12234	SRR716435	NA18620	ERR031877	NA19012	SRR112294		
	SRR397386	NA12249	SRR070525	NA18621	ERR034595	NA19092	SRR100012		
		NA12249	SRR070798	NA18622	ERR032027	NA19093	SRR100033		
70	SRR397210	NA12716	SRR081269	NA18622	ERR032028	NA19098	SRR077453		
	SRR397270	NA12716	SRR081274	NA18623	ERR032008	NA19098	SRR077460		
	SRR397330	NA12717	SRR071172	NA18624	ERR031928	NA19099	SRR748771		
	SRR397390	NA12717	SRR071177	NA18632	ERR031929	NA19099	SRR748772		
		NA12750	SRR077449	NA18633	ERR031878	NA19102	SRR100034		
75	SRR397214	NA12750	SRR081238	NA18635	ERR031879	NA19116	SRR100021		
	SRR397274	NA12750	SRR794547	NA18636	ERR031930	NA19119	SRR077471		
	SRR397334	NA12750	SRR794550	NA18637	ERR031931	NA19119	SRR081271		
	SRR397394	NA12751	SRR071136	NA18853	SRR100011	NA19129	ERR034558		
		NA12751	SRR071139	NA18856	SRR098533	NA19130	SRR107026		
81	SRR397217	NA12760	SRR081223	NA18858	ERR034553	NA19131	SRR070494		

	SRR397277	NA12760	SRR081251	NA18861	ERR034554	NA19131	SRR070783
	SRR397337	NA12761	SRR077753	NA18870	SRR100031	NA19137	SRR081226
	SRR397397	NA12761	SRR081267	NA18871	SRR100029	NA19137	SRR081237
		NA12762	SRR718076	NA18912	SRR111960	NA19137	SRR792542
83	SRR397218	NA12763	SRR077752	NA18940	ERR034596	NA19137	SRR792560
	SRR397278	NA12763	SRR081230	NA18942	ERR034597	NA19138	SRR070472
	SRR397338	NA12812	SRR715913	NA18943	ERR034598	NA19138	SRR070776
	SRR397398	NA12813	SRR718077	NA18944	ERR034599	NA19141	SRR077433
		NA12813	SRR718078	NA18945	ERR034600	NA19141	SRR077464
88	SRR397222	NA12814	SRR715914	NA18947	ERR034601	NA19143	SRR077445
	SRR397282	NA12815	SRR716646	NA18948	ERR034602	NA19143	SRR081272
	SRR397342	NA12872	SRR716647	NA18949	ERR034603	NA19144	SRR077392
	SRR397402	NA12873	SRR702070	NA18951	ERR034604	NA19144	SRR077468
		NA12874	SRR764692	NA18952	ERR034605	NA19152	SRR071135
90	SRR397224	NA12878	SRR098401	NA18953	SRR099546	NA19152	SRR071167
	SRR397284	NA12891	SRR098359	NA18956	SRR766028	NA19153	SRR070660
	SRR397344	NA12892	ERR034529	NA18959	SRR099545	NA19153	SRR070846
	SRR397404	NA18501	SRR100022	NA18960	SRR099533	NA19159	SRR070478
		NA18502	SRR764722	NA18961	SRR099544	NA19159	SRR070786
95	SRR397229	NA18502	SRR764723	NA18964	SRR099539	NA19160	SRR077482
	SRR397289	NA18504	SRR100028	NA18965	SRR764771	NA19160	SRR081250
	SRR397349	NA18505	SRR716648	NA18965	SRR764772	NA19171	SRR077492
	SRR397409	NA18505	SRR716649	NA18966	SRR071175	NA19171	SRR077493
		NA18507	SRR764745	NA18966	SRR071180	NA19172	SRR111962
97	SRR397231	NA18507	SRR764746	NA18967	SRR071192	NA19200	SRR077432
	SRR397291	NA18508	SRR716637	NA18967	SRR071196	NA19200	SRR078847
	SRR397351	NA18508	SRR716638	NA18968	SRR077480	NA19201	SRR077439
	SRR397411	NA18516	SRR100026	NA18968	SRR081231	NA19201	SRR077462
		NA18517	ERR034551	NA18969	SRR081266	NA19204	SRR077857
99	SRR397233	NA18522	SRR107025	NA18969	SRR081273	NA19204	SRR081263
	SRR397293	NA18523	ERR034552	NA18970	SRR071116	NA19206	SRR070491
	SRR397353	NA18526	ERR031854	NA18970	SRR071127	NA19206	SRR070781
	SRR397413	NA18532	ERR031956	NA18971	SRR077447	NA19207	SRR081254

Low-coverage HapMap WGS		CEU					
Sample ID	Run ID	Sample ID	Run ID	Sample ID	Run ID	Sample ID	Run ID
NA06985	SRR400039	NA06985	ERR009159	NA12003	ERR009121	NA12812	ERR009104
NA11832	SRR385763	NA06994	ERR009168	NA12004	ERR009139	NA12813	ERR009114
NA12005	SRR385767	NA07000	ERR009154	NA12005	ERR009155	NA12814	ERR009134
NA12044	SRR393991	NA07051	ERR009147	NA12006	ERR009123	NA12815	ERR009151
NA12760	SRR385773	NA07346	ERR009133	NA12043	ERR009163	NA12872	ERR009099
NA18912	SRR350153	NA07347	ERR009146	NA12044	ERR009157	NA12873	ERR009111
NA18960	SRR442016	NA07357	ERR009167	NA12045	ERR009113	NA12874	ERR009145
NA18968	SRR359062	NA10847	ERR009097	NA12144	ERR009117	NA12891	ERR009105
NA18971	SRR359095	NA10851	ERR009124	NA12154	ERR009129		
NA18974	SRR360136	NA11829	ERR009122	NA12155	ERR009115		
NA18975	SRR359070	NA11830	ERR009140	NA12156	ERR009136		
NA18976	SRR359110	NA11831	ERR009096	NA12234	ERR009144		
NA18981	SRR359083	NA11832	ERR009109	NA12249	ERR009107		
NA18991	ERR052929	NA11840	ERR009142	NA12716	ERR009118		
NA19092	SRR189830	NA11881	ERR009135	NA12717	ERR009164		
NA19119	SRR359106	NA11918	ERR009166	NA12750	ERR009137		
NA19131	SRR359096	NA11920	ERR009149	NA12751	ERR009132		
NA19152	SRR359097	NA11992	ERR009119	NA12760	ERR009130		
NA19171	SRR359061	NA11993	ERR009103	NA12761	ERR009106		
NA19204	SRR359064	NA11994	ERR009141	NA12762	ERR009156		
NA12006	SRR385760	NA11995	ERR009108	NA12763	ERR009152		

Table F.1: Sample ID and Run ID of HLA-typing NGS benchmark dataset.

ID	Covered Variation	Sequence
WT-1		VNSLDPPLLTRYLRI
WT-2		RIHPQSWVHQIALRM
WT-3		LTRYLRIHPQSWVHQ
WT-4		LRIHPQSWVHQIALRM
M-1	L2321F	VNSLDPPLFTRYLRI
M-2	L2321T	VNSLDPPLTTRYLRI
M-3	L2321Y	VNSLDPPLYTRYLRI
M-4	Q2335H	RIHPQSWVHHIALRM
M-5	Q2335H	LRIHPQSWVHHIALRM
M-6	V2333E	LRIHPQSWEHQIALRM
M-7	Y2324L,V2333E	LTRLLRIHPQSWEHQ
M-8	Y2324H,V2333E	LTRHLRIHPQSWEHQ
M-9	R2326K,V2333E	LTRYLKIHPQSWEHQ
M-10	L2321F,V2333E	FTRYLRIHPQSWEHQ
M-11	L2321T,V2333E	TTRYLRIHPQSWEHQ
M-12	L2321Y,V2333E	YTRYLRIHPQSWEHQ
M-13	L2321F,Q2335H	FTRYLRIHPQSWVHH
M-14	L2321T,I2327L	VNSLDPPLTTRYLRL
M-15	I2327L,V2333E	LRLHPQSWEHQIALRM
M-16	L2321T,I2327L,V2333E	LTTRYLRLHPQSWEH

Table F.2: Peptide designs of the de-immunization constructs for experimental evaluation.

H0

ID	Var	Seq	DRB1*03:01	DRB1*07:01	DRB1*15:01	SUM
WT-1		VNSLDPPLLTRYLRI	0.697	0.810	7.274	8.781
WT-2		RIHPQSWVHQIALRM	0.226	40.243	5.029	45.499
WT-3		LTRYLRIHPQSWVHQ	0.243	87.324	30.478	118.045
WT-4		LRIHPQSWVHQIALRM	0.200	56.788	50.927	107.915
M-1	L2321F	VNSLDPPLFTRYLRI	0.165	6.490	0.427	7.082
M-2	L2321T	VNSLDPPLTTRYLRI	0.100	0.100	0.000	0.200
M-3	L2321Y	VNSLDPPLYTRYLRI	0.000	3.600	0.000	3.600
M-4	Q2335H	RIHPQSWVHHIALRM	0.005	75.329	9.756	85.090
M-5	Q2335H	LRIHPQSWVHHIALRM	0.555	65.265	47.784	113.604
M-6	V2333E	LRIHPQSWEHQIALRM	0.157	32.020	44.319	76.496
M-7	Y2324L, V2333E	LTRLLRIHPQSWEHQ	0.026	33.528	5.445	39.000
M-8	Y2324H, V2333E	LTRHLRIHPQSWEHQ	0.000	2.480	13.999	16.480
M-9	R2326K, V2333E	LTRYLKIHPQSWEHQ	1.668	28.951	5.434	36.053
M-10	L2321F, V2333E	FTRYLRIHPQSWEHQ	0.020	43.779	2.906	46.705
M-11	L2321T, V2333E	TTRYLRIHPQSWEHQ	0.544	71.612	6.575	78.731
M-12	L2321Y, V2333E	YTRYLRIHPQSWEHQ	0.155	59.728	3.876	63.758
M-13	L2321F, Q2335H	FTRYLRIHPQSWVHH	1.449	39.809	16.564	57.822
M-14*	L2321T, I2327L	VNSLDPPLTTRYLRL				
M-15	I2327L,V2333E	LRLHPQSWEHQIALRM	0.032	42.841	18.099	60.972
M-16	L2321T, I2327L, V2333E	LTTRYLRLHPQSWEH	0.236	56.611	8.924	65.771

H24

ID	Var	Seq	DRB1*03:01	DRB1*07:01	DRB1*15:01	SUM
WT-1		VNSLDPPLLTRYLRI	0.000	0.370	0.000	0.370
WT-2		RIHPQSWVHQIALRM	0.000	35.522	0.003	35.524
WT-3		LTRYLRIHPQSWVHQ	0.243	74.002	15.728	89.973
WT-4		LRIHPQSWVHQIALRM	0.021	38.252	17.219	55.491
M-1	L2321F	VNSLDPPLFTRYLRI	0.165	0.497	0.000	0.662
M-2	L2321T	VNSLDPPLTTRYLRI	0.000	0.000	0.000	0.000
M-3	L2321Y	VNSLDPPLYTRYLRI	0.000	0.900	0.000	0.900
M-4	Q2335H	RIHPQSWVHHIALRM	0.000	53.300	0.000	53.300
M-5	Q2335H	LRIHPQSWVHHIALRM	0.000	40.700	20.400	61.100
M-6	V2333E	LRIHPQSWEHQIALRM	0.000	19.800	11.600	31.400
M-7	Y2324L, V2333E	LTRLLRIHPQSWEHQ	0.000	20.100	0.100	20.200
M-8	Y2324H, V2333E	LTRHLRIHPQSWEHQ	0.000	0.100	2.000	2.100
M-9	R2326K, V2333E	LTRYLKIHPQSWEHQ	0.000	17.900	0.300	18.200
M-10	L2321F, V2333E	FTRYLRIHPQSWEHQ	0.000	27.400	0.300	27.700
M-11	L2321T, V2333E	TTRYLRIHPQSWEHQ	0.000	49.600	1.600	51.200
M-12	L2321Y, V2333E	YTRYLRIHPQSWEHQ	0.200	40.700	0.700	41.600
M-13	L2321F, Q2335H	FTRYLRIHPQSWVHH	0.000	29.800	7.500	37.300
M-14*	L2321T, I2327L	VNSLDPPLTTRYLRL				
M-15	I2327L,V2333E	LRLHPQSWEHQIALRM	0.000	21.500	5.300	26.800
M-16	L2321T, I2327L, V2333E	LTTRYLRLHPQSWEH	0.236	13.754	8.924	22.914

* Failed synthesis

Table F.3: Experimentally determined immunogenicity scores of designed peptides generated by a commercial REVEAL HLA-peptide binding assay of ProImmune (`www.proimmune.com`). The measurements were taken at the start of incubation (H0) and after 24 hours (H24).

Variant ID	Amino acid (HGVS)	Amino acid (Legacy)	Protein Change	Severity	Inhibitors
1682	2200	2181	E2200D	Mild	Yes
1685	2204	2185	I2204T	Mild	No
1691	2209	2190	I2209N	Moderate	
1692	2211	2192	A2211P	Moderate	No
1711	2237	2218	A2237T	Mild	Yes
1723	2247	2228	E2247D	Mild	Yes
1724	2248	2229	W2248S	Moderate	
1726	2249	2230	L2249R	Severe	Yes
1728	2250	2231	Q2250H	Mild	No
1729	2251	2232	V2251E	Severe	
1737	2264	2245	T2264A	Mild	
1741	2266	2247	G2266R	Severe	No
1743	2266	2247	G2266E	Severe	No
1745	2272	2253	T2272P	Mild	No
1746	2272	2253	T2272P	Mild	No
1747	2274	2255	M2274K	Mild	No
1751	2276	2257	V2276G	Severe	Yes
1754	2279	2260	F2279C	Severe	Yes
1753	2279	2260	F2279S	Severe	No
1755	2280	2261	L2280P	Severe	No
1756	2281	2262	I2281T	Severe	
1757	2284	2265	S2284R	Mild	No
1760	2290	2271	W2290L	Moderate	
1771	2302	2283	F2302V	Moderate	No
1772	2302	2283	F2302S	Severe	No
2135	2310	2291	T2310P	Mild	No
1784	2314	2295	N2314Y	Mild	No
1790	2323	2304	R2323G	Mild	No
1793	2323	2304	R2323L	Mild	No
1796	2326	2307	R2326G	Severe	No
2136	2329	2310	P2329S	Mild	No
1801	2329	2310	P2329L	Severe	No
1807	2332	2313	W2332S	Moderate	No
1815	2339	2320	R2339M	Moderate	No
1820	2343	2324	L2343P	Mild	No
1823	2344	2325	G2344C	Moderate	No
1821	2344	2325	G2344S	Severe	
1825	2344	2325	G2344A	Mild	No
1824	2344	2325	G2344D	Severe	
1826	2345	2326	C2345Y	Severe	

Table F.4: Hemophilia A severity data for single point mutations within the C2 domain extracted from factor VIII variant database (`http://www.factorviii-db.org`).

Method	Version	Usage	Platform Compatibility	Reference
HLA binding:				
SYFPEITHI	1.0	T-cell epitope	Windows, Linux, Mac	(Rammensee *et al.*, 1999) [111]
BIMAS	1.0	HLA-I binding	Windows, Linux, Mac	(Paerker *et al.*, 1994) [113]
SVMHC	1.0	HLA-I binding	Windows, Linux, Mac	(Dönnes *et al.*, 2002) [241]
ARB	1.0	HLA-I binding	Windows, Linux, Mac	(Bui *et al.*, 2005) [242]
SMM	1.0	HLA-I binding	Windows, Linux, Mac	(Pepters *et al.*, 2005) [112]
SMMPMBEC	1.0	HLA-I binding	Windows, Linux, Mac	(Kim *et al.*, 2009) [243]
Epidemix	1.1	HLA-I binding	Windows, Linux, Mac	(Feldhahn *et al.*, 2009) [207]
Comblib Sidney 2008	1.0	HLA-I binding	Windows, Linux, Mac	(Sidney *et al.*, 2008) [244]
PickPocket*	1.1	HLA-I binding	Linux, Mac	(Zhang *et al.*, 2009) [245]
NetMHC*	3.0, 3.4, 4.0	HLA-I binding	Linux, Mac	(Lundegaard *et al.*, 2008) [225]
NetMHCpan*	2.4, 2.8, 3.0	HLA-I binding	Linux, Mac	(Hoof *et al.*, 2009) [246]
HAMMER	1.0	HLA-II binding	Windows, Linux, Mac	(Sturniolo *et al.*, 1999) [247]
TEPITOPEpan	1.0	HLA-II binding	Windows, Linux, Mac	(Zhang *et al.*, 2012) [167]
NetMHCII*	2.2	HLA-II binding	Linux, Mac	(Nielsen *et al.*, 2007) [248]
NetMHCIIpan*	3.0, 3.1	HLA-II binding	Linux, Mac	(Karosiene *et al.*, 2013) [249]
UniTope	1.0	T-cell epitope	Windows, Linux, Mac	(Toussaint *et al.*, 2011) [250]
NetCTLpan*	1.1	T-cell epitope	Linux, Mac	(Stranzl *et al.*, 2010) [251]
Cleavage Prediction:				
PteaSMM (C/S20)	1.0	Cleavage site	Windows, Linux, Mac	(Tenzer *et al.*, 2005) [115]
PCM	1.0	Cleavage site	Windows, Linux, Mac	(Dönnes *et al.*, 2005) [114]
NetChop*	3.1	Cleavage site	Linux, Mac	(Nielsen *et al.*, 2005) [252]
Ginodi	1.0	Cleavage fragment	Windows, Linux, Mac	(Ginodi *et al.*, 2008) [253]
TAP Prediction:				
SVMTAP	1.0	TAP affinity	Windows, Linux, Mac	(Dönnes *et al.*, 2005) [114]
SMMTAP	1.0	TAP affinity	Windows, Linux, Mac	(Peters *et al.*, 2003) [254]
Additive matrix method	1.0	TAP affinity	Windows, Linux, Mac	(Doytchinova *et al.*, 2004) [255]
Epitope Selection:				
OptiTope+	1.0	Epitope selection for vaccine design	Windows, Linux, Mac	(Toussaint and Kohlbacher, 2009) [223]
Epitope Assembly:				
TSP approach+	1.0	String-of-beads design	Windows, Linux, Mac	(Toussaint *et al.*, 2011) [101]
Spacer design + TSP+	1.0	Spacer design	Windows, Linux, Mac	(Schubert and Kohlbacher, 2016) [256]
HLA Typing:				
OptiType*	1.0	HLA-I typing	Linux, Mac	(Szolek *et al.*, 2014) [77]
Polysolver*	1.0	HLA-I typing	Linux, Mac	(Shukla *et al.*, 2015) [88]
Seq2HLA*	2.2	HLA-I/II typing	Linux, Mac	(Bögel *et al.*, 2013) [70]
ATHLATES*	1.0	HLA-I/II typing	Linux, Mac	(Liu *et al.*, 2013) [71]

* Installation of external software is required.

+ An integer linear programming solver such as CBC (`https://projects.coin-or.org/Cbc`) is required. For epitope assembly the LKH approximation software (`http://www.akira.ruc.dk/~keld/research/LKH`) is advised to use.

Table F.5: Supported prediction methods of FRED 2.

Epitope	HLA-Ligand	T-Cell reactive	IEDB ID
FMYSDFHFI	Yes	Yes	17119
MMMGMFNML	Yes	Yes	42143
YLMAWKQVL	Yes		124888
FVANFSMEL	Yes	Yes	97314
WMMAMRYPI	Yes		124859
FLARSALIL	Yes	No	16522
FMQALQLLL	Yes		178842
LLIDGTASL	Subsequence	Yes	129079, 129607, 212044, 218205
ATYQRTRAL	Yes	Yes	5230, 7655, 41793, 79763, 146073, 164384, 181194
FVRQCFNPM	Yes		18274

Table F.6: Experimental evidence for the *in silico* predicted and selected epitopes and their corresponding IEDB IDs.