Computational immunogenetics in allogeneic immunotherapy

Von der Naturwissenschaftlichen Fakultät

der Gottfried Wilhelm Leibniz Universität Hannover

zur Erlangung des Grades eines

DOKTORS DER NATURWISSENSCHAFTEN

Dr. rer. nat. –

genehmigte Dissertation

von

M.Sc. David S. DeLuca

geboren am 02. Juli 1980 in Pittsfield, MA, USA

2008

Referent: Prof. Dr. Scheper Institut für Technische Chemie Gottfried Wilhelm Leibniz Universität Hannover

Koreferent: Prof. Dr. Blasczyk Institut für Tranfusionsmedizin Medizinische Hochschule Hannover

Tag der Promotion: 30.07.2008

This work was produced in the Institut für Technische Chemie at the Gottfried Wilhelm Leibniz Universität Hannover under the direction of Prof. Dr. T. Scheper and in cooperation with Prof. Dr. R. Blasczyk of the Institut für Transfusionsmedizin, Medizinische Hochschule Hannover.

– to my family –

– meiner Familie gewidmet –

Acknowledgements

I am sincerely grateful for the support that I have received over the course of my doctoral training. I would like to acknowledge the support of Prof. Scheper and Prof. Hitzmann from the Leibnitz Universiät Hannover.

From Hannover Medical School, I would like to acknowledge the following people. All of the goals achieved in this time period were those chosen by my mentor, Prof. Dr. Rainer Blasczyk, and completed under his direction. I am also grateful for the scientific guidance of Prof. Holgar A. Elsner and P.D. Dr. Peter A. Horn. For the PeptideCheck program in particular, Prof. Britta Eiz-Vesper provided productive discussions as well as experimental results. The assistance of my colleague, Nektarios Ladas, has contributed dramatically to the quality of the technical implementation of the software and database systems described in this dissertation. I am particularly grateful for the deeper understanding of database optimization which he has imparted upon me. Dr. Christina Bade-Doeding has been an important partner for investigation of protein/peptide interactions, and has provided experimentally determined peptide sequences eluted from soluble HLA. The assistance on mathematical issues provided by Dr. Wolfgang Mende was very valuable.

Further external cooperation partners contributed to this work. Yoram Louzoun of Bar Ilan University, Israel provided assistance in integrating his PepCleave algorithm in our systems. The MHCBN database was kindly and conveniently provided by GPS Raghava and Manoj Bhasin of the Institute of Microbial Technology, Chandigarh, India. The NetMHC algorithm and technical support were provided by Søren Brunak and Claus Lundegaard of the Center for Biological Sequence Analysis, BioCentrum- DTU, Denmark.

Finally, I would like to thank the cooperation partners of the StemNet project from the University of Jena for introducing me to the concept of Ontology and its utility in bioinformatics. The production of the MHC Ontology was undertaking in close cooperation with Elena Beisswanger, Joachim Wermter, and Prof. Udo Hahn.

Zusammenfassung

Immungenetische Informatik in der allogenen Immuntherapie

Der Bereich Immungenetik ist ein Gebiet, das in hohem Ausmaß von den Organisations- und Analyse-Möglichkeiten der Bioinformatik profitieren kann. In der hämatopoietischen Stammzelltransplantation spielen die funktionelle Wirkungen von genetischen Polymorphismen eine bedeutende Rolle. Hier führt eine genetische Inkompatibilität zum Krankheitsbild der Graft versus Host Disease. Auf der anderen Seite sind diese Variationen für die gewünschten Graft versus Leukemia Effekte verantwortlich.

In dieser Arbeit werden die Wirkungen solcher Polymorphismen innerhalb und auch außerhalb des Major Histocompatiblitätskomplexs (MHC) computertechnisch analysiert, um ihre funktionelle Bedeutung vorherzusagen. Diese Vorgehensweise erfordert das Vernetzen von öffentlichen Gen-, Protein-, und Polymorphismen-Datenbanken. Darüber hinaus werden die physiochemischen Eigenschaften von Aminosäuren berücksichtigt, um die Strukturähnlichkeiten von Proteinen zu quantifizieren. Um MHC-Peptidinteraktionen vorherzusagen, wurden Peptidbindungsdaten statistisch analysiert. Wegen des großen Datenvolumens wurden die Prinzipien des Data Warehousing angewendet.

Die resultierende Systeme bieten die Fähigkeit:

1. Strukturähnlichkeiten zwischen zwei MHC Proteinvarianten zu quantifizieren und zu qualifizieren,
2. MHC-Peptidinteraktionen vorherzusagen,
3. die MHC Vielfalt so zu organisieren, dass die Anzahl der MHC Varianten, für die eine Peptidbindungsvorhersage möglich ist, erweiterbar ist, und
4. Peptidziele für immuntherapeutische Anwendungen gegen Leukemie zu identifizieren, die keine Graft versus Host Disease hervorrufen.

Alle Programmanwendungen sind online unter www.peptidecheck.org zu erreichen.

Stichwörte:

Major Histocompatiblitätskomplex, Bioinformatik, hämatopoietischen Stammzelltransplantation

Summary

Computational immunogenetics in allogeneic immunotherapy

The field immunogenetics is an area which can benefit greatly from the possibilities offered by computational biology for data organization and analysis. The functional consequences of genetic polymorphisms play a major role in hematopoietic stem cell transplantation, and other forms of immunotherapy. Here, genetic incompatibility can lead to the complication known as graft versus host disease. However, such genetic variations are also responsible for the beneficial graft versus leukemia effect.

In this work, the effects of polymorphism inside and outside of the major histocompatibility complex (MHC) are analyzed computationally to predict their functional significance. This approach requires the networking of public gene, protein, and polymorphism databases. Furthermore, the physicochemical properties of amino acids were considered to quantify the structural similarity of protein variants. To predict MHC-peptide interactions, peptide binding databases were statistically analyzed. Because of the volume of data processed, the principals of data warehousing had to be applied.

The resulting systems provide the ability to

1. quantify and qualify the structural similarity between two MHC protein variants,
2. predict MHC-peptide interaction,
3. manage MHC diversity to expand the number of variants for which peptide binding prediction is possible, and
4. identify anti-leukemia peptide targets for immunotherapeutic application without causing graft versus host disease.

All of these programs are accessible as web tools and are available online at www.peptidecheck.org.

Keywords:

major histocompatibility complex, bioinformatics, hematopoietic stem cell transplantation

Contents

Abbreviations

CML	chronic myeloid leukemia
GvHD	graft versus host disease
GvL	graft versus leukemia
HLA	human leukocyte antigen
HSCT	hematopoietic stem cell transplantation
HTTP	hypertext transfer protocol
mHag	minor histocompatibility antigen
MHC	major histocompatibility antigen
OWL	web ontology language
SNP	single nucleotide polymorphism
TCR	T cell receptor
XML	extensible markup language

Part I. Preface

In accordance with the standards set by the Leibnitz Universität Hannover, this cumulative dissertation is a collection of articles which were published in, or prepared for established scientific journals with strict peer review systems. In addition to four original papers, I have included two book contributions and a review article. All articles concern computational approaches to analyzing the genes of the major histocompatibility complex and the protein interactions which are determined by this gene region. The work presented in each article was undertaken with the goal of furthering the development of immunotherapeutic solutions to the problem of leukemia.

The articles in this dissertation are preceded by a cumulative analysis (Part II), which aims to generalize and summarize the issues, results, and conclusions found within the articles. The Appendix has provided me with an opportunity to describe computer science concepts and implementations which were important for this work, but could not be included in the articles themselves. As a result, the Appendix is not simply a repository for supplementary tables, but also includes detailed descriptions the how particular programming concepts contributed to the performance and design of the systems. However, because these issues are of a technical nature, and do not involve the natural sciences, I feel that this text is appropriately placed in the Appendix, as opposed to in the main body.

This cumulative dissertation includes the following articles. Original papers are given in bold face. A description of to what extent I personally contributed is given after each article in brackets.

- DeLuca, D.S. and Blasczyk, R. (2007) The immunoinformatics of cancer immunotherapy, Tissue Antigens, 70, 265-271. [Review, concept and text by DeLuca.]

- **Elsner, H.A., DeLuca, D., Strub, J. and Blasczyk, R. (2004) HistoCheck: rating of HLA class I and II mismatches by an**

internet-based software tool, Bone Marrow Transplant, 33, 165-169. [Concept and text by Elsner. Labor by DeLuca.]

- DeLuca, D.S. and Blasczyk, R. (2007) HistoCheck: Evaluating Structural and Functional MHC Similarities. In Flower, D.R. (ed), Immunoinformatics. Humana Press, 395-405. [Concept and text by DeLuca.]

- **DeLuca, D.S., Khattab, B. and Blasczyk, R. (2007) A modular concept of HLA for comprehensive peptide binding prediction, Immunogenetics, 59, 25-35.** [Concept by Blasczyk. Text and labor by DeLuca.]

- DeLuca, D.S. and Blasczyk, R. (2007) Implementing the Modular MHC Model for Predicting Peptide Binding. In Flower, D.R. (ed), Immunoinformatics. Humana Press, 261-271. [Concept and text by DeLuca.]

- **DeLuca, D.S., Eiz-Vesper, B., Ladas, N., Khattab, B., and Blasczyk, R. (2008) High thoughput minor histocompatibility antigen prediction, Bioinformatics (to be submitted).** [Concept by Blasczyk. Text by DeLuca. Labor by DeLuca (70%).]

- **DeLuca, D.S., Beisswanger, E., Wermter, J., Horn, P.A., Hahn, U., Blasczyk, R. (2008) Development and immunoinformatic application of the MHC ontology, Bioinformatics (to be submitted).** [Concept and text by DeLuca. Labor by DeLuca (80%).]

Although the original papers are sufficient to meet the requirements of this dissertation, I felt it was important to include the book contributions and the review as well. The review article in *Tissue Antigens* provides a useful introduction to the issues surrounding immunotherapy. Furthermore, it establishes the state-of-the-art of minor histocompatibility antigen prediction as a prelude to the article prepared for *Bioinformatics* on the same topic. For the topics of HistoCheck and the modular concept of HLA, each original paper is followed by a book contribution. The book contributions were published as part of the series, *Methods in Molecular Biology*, and supplement the original papers by going into more detail about methods and implementation. The final paper involves the development of an ontology, to which our cooperation partners at the University of Jena contributed greatly. Here, we transferred a concept from computer linguistics to the area of immunogenetics to improve the design of the systems described in the other papers. In summary, all articles were selected with care to contribute to this dissertation in a cohesive and synergetic manner.

Part II. Cumulative Analysis

1 Introduction

The work presented here combines computer science technologies with biological concepts for the purpose of furthering understanding of the immune processes involved in hematopoietic stem cell transplantation (HSCT).

1.1 Clinical and Immunogenetic Background

Hematopoietic Stem Cell Transplantation

HSCT is perhaps the most clinically relevant form of immunotherapy currently practiced, second only to vaccination. In treating hematopoietic malignancies, irradiation of the patient's bone marrow is performed with the goal of eliminating cancerous cells, and with the side effect of destroying healthy stem cells. Following irradiation, HSCT replenishes these stem cells, providing the patient with, in effect, a donated immune system. The issue of histocompatibility then plays a central role in the three main problems faced by the patient: rejection, graft versus host disease (GvHD), and relapse. Directly following engraftment, there is a risk that the graft is unable to establish itself because the native immune system reacts with antigens presented on the surface of graft cells. When the graft is established, the donated immune cells cause an immune response to the patient's tissues, in particular liver, skin, and intestinal cells. This is the cause of GvHD. If the donated immune system is not able to eliminate residual leukemic cells, then the patient goes into relapse. GvHD and relapse have a kind of inverse relationship. When genetic polymorphisms lead to a significant presence of reactive antigen in the patient, then the risk of GvHD is higher. However, higher levels of antigen contribute to the graft's ability to react with residual leukemia cells, preventing relapse[1]. This positive immune reaction is known as the graft versus leukemia (GvL) effect. The Holy Grail of HSCT is to establish a therapy which favors the GvL reaction and limits GvHD.

Antigen-Targeted Immunotherapy

In the context of a HSCT, it is possible to stimulate the donated T cells *ex vivo* prior to implantation in the patient. This is known as adoptive transfer[2, 3, 4]. The purpose of adoptive transfer is to apply antigen to T cells prior to implantation for the purpose of causing antigen-specific proliferation, with the hope of eliciting a specific immune response within the patient. The efforts of this dissertation concern anti-leukemia responses, although it can be noted that anti-pathogen responses are also being pursued by other groups. Because of the threat of GvHD, immune responses specific to healthy patient tissues should be avoided through the careful selection of antigen. Furthermore, donated T cells will be tolerant to antigens which are present on the surface of donor cells. Therefore, to elicit the desired immune response, antigens must be selected which are absent in the donor, but present in the patient. Differences in the antigen profiles between donor and patient can be cause by single nucleotide polymorphisms (SNPs). Antigens having these characteristics which are presented by proteins encoded by the major histocompatibility complex (MHC) are known as minor histocompatibility antigens (mHag) [5] and are introduced below.

Major and Minor Histocompatibility Antigens

The major histocompatibility complex (MHC) is a gene complex located on chromosome six in humans, which had been identified early on as a determinant for compatibility following tissue transplantation. The genes found here encode proteins which are able to bind peptide fragments within the cell and transport them to the cell surface. On the cell surface, the MHC-peptide complex is available for binding by T lymphocytes (a.k.a. T cells) which can initiate an immune response, leading to the destruction of the target cell, and the activation of further lymphocytes. T cells express a protein on their cell surface known as the T cell receptor (TCR), whose structure is variable among T cells. The interaction between the TCR and the MHC-peptide complex determine whether the immune response is initiated or not. To prevent T cells from reacting with normal tissues, they mature in the thymus, where self-reactive T cells are eliminated. In this way, T cells normally react only with foreign structures presented in the MHC.

In humans MHC is known as human leukocyte antigen (HLA). HLA has a high rate of polymorphisms. To date, there are over 3000 alleles known to be coded among about a dozen highly-investigated genes[6]. These polymorphisms lead to structural differences in the HLA proteins which cause immune reactivity following HSCT. The TCRs of the donated T cells were selected in the donor's thymus to be tolerant of the HLA structures of the donor. If the patient expresses a different HLA structure, then the donated T cells cause an immune reaction. If the patient and donor are HLA identical, then there is less chance of a reaction. However, polymorphisms in other loci can lead to a differential expression of peptides which are presented by HLA. In this case, donated T cells will not reactive negatively to the HLA molecule itself, but to the peptides presented within. Such peptides are designated minor histocompatibility antigens (mHag)[7].

1.2 Bioinformatic Starting Block

This dissertation covers computational techniques for analyzing genetic polymorphisms within and beyond the MHC for the purpose predicting their functional significance. The following paragraphs are meant to describe the resources available for taking on this task.

The nucleotide sequences of alleles within the HLA are provided by public databases[6]. These sequences are translated algorithmically to provide protein sequences. Understanding the function and significance of amino acids at particular positions is of great importance for predicting immune reactions. Fortunately, x-ray crystallography has provided immunogeneticists with the 3-dimensional structure of over a hundred HLA protein variants[8]. These models are extremely informative with respect to the structure and function of all amino acid positions within the HLA molecule.

The interaction of HLA proteins with the peptides they bind is determined by physiochemical environment provided in the structural domain known as the peptide binding domain (or groove). To identify which positions in the protein sequence are directly involved in peptide binding, the crystallographic data has been mathematically analyzed, either via solvent accessibility calculations, or simple atomic distance calculations[9]. The sub domains within the peptide binding groove which are responsible for binding the individual amino acids of the peptide are known as pockets. These pocket definitions are a further important resource for making further predictions about the effects of variation on HLA function.

Another indispensable source of information for analyzing HLA-peptide interaction is the results of peptide elution and sequencing experiments. These experiments have been performed by the greater immunogenetics community, and compiled into large databases[10, 11, 12, 13]. For example, there are over a thousand peptide sequences which have been determined to bind HLA-A*0201. Inspection of these peptide sequences reveals certain motifs. Each HLA variant has a particular binding specificity, and motif preference[14]. This phenomenon allows us to utilize these data to create prediction algorithms to predict HLA binding for a given peptide.

In the case of mHags, we must look beyond the polymorphism in HLA and consider any polymorphism throughout the genome which can lead to differential peptide expression. Single nucleotide polymorphism (SNP) data are provided to the public by the dbSNP hosted at NCBI, NIH (USA). These polymorphisms can be found in coding or non-coding regions. They can lead to an amino acid exchange (non-synonymous), or be silent. They can cause frame shifts via nucleotide insertion or deletion. They can destroy a wild type stop codon, or cause the addition of a premature stop codon. For investigating mHags, these data must be filtered to find only those mutations which can lead to a protein difference. The SNP data are linked via protein Ids to protein sequences in the public Entrez Protein database. By combining this information, it is possible to produce a database of polymorphic peptide sequences. This is exactly what we did, as reported in Section 9.

In addition to the SNP requirements, mHags must be processed by the proteasome, and then bound by HLA proteins and presented on the surface of the cells. Analysis by in vitro digestion assays has provided a source of experimental data for the bases of proteasomal cleavage prediction algorithms[15, 16, 17, 18, 19, 20]. In addition, HLA peptide binding prediction has been long studied, and represents one of the oldest applications of computational biology in the area of immunogenetics[21, 22, 23, 24, 25, 26, 27, 20]. These prediction algorithms are essential for taking on the challenge of prediction mHag peptides.

1.3 Challenges and Objectives

For the immunotherapeutic treatment of leukemia, genetic polymorphism is simultaneously a barrier and an opportunity. Polymorphisms within the MHC lead to graft rejection and GvHD. Polymorphism outside of the MHC can lead to GvHD, but also to the destruction of leukemia cells (GvL). Immunogeneticists are posed with the challenge of interpreting polymorphism typing results for individual donor/patient pairs. The experimental data, which could provide insight into the immune processes, is currently in the form of thousands of nucleotide sequences of HLA alleles, thousands of amino acid sequences of HLA bound peptides, and millions of reported polymorphisms throughout the genome. Examining these data by hand is impossible due to the sheer volume. Computer systems offer a chance to overcome this problem.

The objective of this dissertation is to utilize computer algorithms and databases to examine how polymorphism affects immunogenicity in the context of immunotherapy. The ultimate goal is to identify leukemia-specific antigens which can be utilized in immunotherapy to eliminated leukemia without causing GvHD. This endeavor requires the ability to

1. quantify and qualify the structural similarity between two MHC protein variants,
2. predict MHC-peptide interaction,
3. manage MHC diversity to expand the number of variants for which peptide binding prediction is possible, and
4. identify anti-leukemia peptide targets for immunotherapeutic application without causing graft versus host disease.

Each step is a prerequisite for the next. Comparing structural and functional similarity of MHC protein variants is required in order to understand how amino acid differences affect peptide binding. By doing this, in combination with predicting MHC-peptide binding in general, we aim to expand the number of variants for which peptide binding prediction is possible. This ensures that clinical applications can be individualized, and not developed exclusively for carriers of the most common variants. Because MHC-peptide binding is a critical step in immune reactivity during immunotherapy, these predictions contribute to the ultimate goal of identifying leukemia specific antigens.

2 Results and Discussion

The result of these efforts is a collection of algorithms, databases, and web servers which provide insight into the functional significance of HLA and non-HLA polymorphisms for immune reactivity.

2.1 HLA Protein Structure

The web-based HistoCheck program was created for the purpose of comparing HLA alleles to evaluate their functional similarity (see Sections 5 and 6). The motivation for this program is that clinicians are often faced with a transplantation situation in which an HLA mismatch is unavoidable. Here, the question is: Which mismatch is preferable? HistoCheck utilizes the protein sequence information from the IMGT/HLA database to identify which amino acids differ between the two variants. For each amino acid mismatch, HistoCheck evaluates both the functional relevance of the position of the mismatch, as well as the physicochemical similarity of the amino acids themselves.

The functional relevance of each position in the HLA protein was provided by crystallographic data[8]. Here, two functions were considered: peptide binding, and TCR interaction. As a result, a given position was tagged as PEP (peptide binding), TCR (interacting with TCR), or PEP/TCR (having both functions). Positions having neither function were not tagged. While the position of an amino acid exchange is critical to the effect, the characteristics of the amino acids involved also play an important role. For example, an amino acid exchange involving two physiochemically similar residues is likely to have little effect upon the function of HLA protein. To quantify amino acid similarity, Risler scores are provided to the user by HistoCheck. These scores are based upon the rate of amino acid substitutions among evolutionarily related proteins. A low score reflects a high rate of substitution among related proteins, and therefore a high level of functional similarity. Conversely, high scores represent low similarity. An algorithm was proposed to summarize and quantify the extent of functional variation. This algorithm is defined and explained in the Methods of Section 5. For visualization, a 3-dimensional model of the HLA protein is provided and the relevant positions are highlighted. Instructions on how to interpret the data provided by HistoCheck are given in Section 6.

Whether there is a direct correlation between the final score given for two HLA alleles and the level of GvHD following HSCT has not been determined. While this detracts greatly from the expressiveness of this score, it does not degrade the overall value of the HistoCheck website. In particular, the primary data given to the user (position of mismatches, amino acids involved), as well as the visualization on the crystallographic structure provide practical information about an HLA mismatch. The final interpretation is left up to the user. For the following work in this dissertation, the experience of implementing the HistoCheck website was essential because the programming methods and data structures involved provided a stepping stone to the implementation of the modular concept of HLA.

2.2 HLA Peptide Binding

Programming HistoCheck was the first step in analyzing the effects of polymorphisms on HLA protein structure. The next step was to investigate how these variations affect peptide binding. Here, databases of peptide sequences from experimentally determined HLA binders were very important. By utilizing a published peptide sequence database[10], a prediction algorithm was generated based upon the frequencies of amino acids at each position in the peptide. This was done for each HLA protein for which enough peptide binding data was available (more than 15 peptides per HLA variant). However, because of the high costs of eluting and sequencing peptides, we went further to develop an algorithm which considers the structural similarities among HLA proteins and exploits these similarities to expand the number of HLA variants for which binding prediction is possible. This was dubbed, a modular concept of HLA (see Sections 7 and 8).

By using the x-ray crystallography-based pocket definitions, we generated a database of HLA "modules", which consist of non-sequential lists of amino acids which represent the physiochemical environment for a given position in the bound peptide. To make a prediction for a given HLA protein, the peptide binding data for each module was utilized. We then demonstrated that it is possible to make an accurate prediction for an HLA protein variant, even if there is no peptide binding data available for this variant.

The accuracy of the predictions was measured using by calculating the area under the receiver operating characteristic curve (A_{ROC}). The ROC curve is the relation between the sensitivity and specificity of the prediction algorithm (see Materials and methods, Section 7). The algorithms sensitivity and specificity are measured by the so-called jackhammer technique. Here, the entire set of known binding peptides is used to train the prediction algorithm, excluding peptide reserved for testing, and any peptide of high similarity. The testing peptide is then applied to the prediction algorithm to evaluate accuracy. This is repeated to test with every peptide in the training/testing set, and ensures that the testing and training datasets are strictly separated.

The resulting A_{ROC} values were very high for both module-based and matrix-based peptide binding prediction (see Table 3 in Section 7), confirming the accuracy of this approach. Furthermore, Figure 2 of Section 7 demonstrates that highly accurate predictions can be made using the modular approach for alleles for which no peptide binding data are available. Whereas the standard approach to HLA peptide prediction could only make predictions for 28 alleles using the given data, the modular technique was able to increase this number to 144. These results bring us one step closer to providing complete population coverage for individualized immunotherapies which rely on such predictions.

The generation of HLA modules provided insight into the nature of HLA diversity. A total of 2,525 modules were created for 1,098 Class I alleles. Because there are nine modules for each allele, this represents a 71% conservation of modular sequences. Furthermore, the level of module diversity differs greatly from pocket to pocket (see Figure 1, Section 7). For example, only 82 modules were generated for pocket 8, which has a minimal effect in determining the peptide binding specificity. In contrast, for pocket 6, which is considered to be an “anchor position” (i.e. highly significant for determining specificity), 458 modules were generated.

The modular concept of HLA provided a new way to approach the question of which alleles should be considered when acquiring new peptide binding data. Three systems of ranking HLA variants were proposed. The variants are ranked based upon how much information they deliver to the module-based system of peptide binding prediction when the bound peptides are determined. By utilizing this ranking system, resources can be utilized as efficiently as possible. Firstly, alleles were ranked based upon the number of new modules having peptide binding data which would be entered into the system. A second ranking system was based upon the number of new anchor modules (modules for pockets 2 and 9) with associated binding data. Finally, perhaps the most easily to understand ranking is that which is based upon the number of new alleles for which prediction would be possible when the peptide binding data is incorporated. The results of these ranking schemas are given in Table 5 of Section 7. For example, the determination of the peptide binding motif for B*4808 alone would allow for the peptide binding prediction of 16 additional alleles

2.3 Minor Histocompatibility Antigens

The ability to predict HLA-peptide binding was an important prerequisite for the investigation of mHags. This comes from the obvious fact that mHags are HLA-bound peptides. However, these peptides must fulfill additional requirements in order to be functional mHags: they must be polymorphic, they must be naturally processed by the proteasome, and they must cause an immune response in an allogeneic transplantation setting. To simulate this, we created the PeptideCheck web resource (Section 9). To use a term from the world of IT (Information Technology), we created a *data warehouse* to manage all of the biological data. A schematic is given in Appendix A.

The source of polymorphism data was the dbSNP hosted by NCBI. NCBI provides so-called eUtilities, which allow programmers to access data in an automated fashion, using the HTTP web protocol. Since we are only interested in polymorphisms which lead to an amino acid difference, it was possible to utilize the eUtilities query to pre-filter the data in this regard. The data was downloaded in XML format, stored temporarily, and then processed and reorganized into database tables using a combination of Java and Caché Object Script. As given in Table 1 of Section 9, almost 49,000 SNP entries were collected, of which almost 23,000 were listed as validated. The significance of the validation label was underscored, when we performed SNP typing on selected candidates (Table 5, Section 9). Strikingly, zero of 5 non-validated SNPs could be confirmed. In contrast, 4 of 6 validated SNPs could be confirmed in our lab.

The SNP data was used in combination with almost 24,000 protein sequences to create a database of almost 2 million allogeneic peptide candidates. Prediction algorithms, such as those from Section 7, as well algorithms from other groups, including proteasomal processing algorithms were applied to all peptides. The resulting system provides a method of querying peptides to find those which fulfill the genetic, polymorphic, and functional requirements necessary to be considered mHags.

Comparison to Validated mHags

To validate this system, the mHag candidates were compared to the currently known, experimentally determined mHags described in dbMinor[28]. The PeptideCheck ranking system was able to reproduce the experimental results, ranking three of the known mHags in the top 0.25 % of possible peptide candidates: HA-1, HA-3, and HA-8. Remarkably, this ranking scheme placed HA-3 at position 2 of over 800,000 candidates.

Not all of the known mHags from the dbMinor could be reproduced by the PeptideCheck system (see Table 2, Section 9). Of the 21 polymorphism known to produce mHags, 5 were types of polymorphism which are not easily reproduced computationally, or are simply not available. For example, several mHags are caused by gene deletion, for which there is currently no data available. Some mHags are caused by SNPs which are listed in dbSNP, but occur in non coding regions, and result in differential peptide expression via mechanisms such as alternative splicing. Such phenomena are difficult to compute, and thus not included in PeptideCheck. Two of the known mHags are caused by SNPs which were never reported to the dbSNP, and therefore excluded from our system. A final limitation of PeptideCheck is that only peptides having nine amino acids are considered. For a long time, this length was considered to be the canonical length of peptides, and the majority of reported HLA-bound peptides are nonamers. As a result, HLA peptide binding prediction algorithms, such as the ones developed during this dissertation, often focus only on nonamers. However, many of the known mHags vary in length. The prediction of variable-length peptides would be an important improvement for the further development of PeptideCheck.

GvL Targeted mHags

Because our goal was to find targets for immunotherapy, the system utilizes tissue expression to provide a list of GvL-inducing peptides which are unlikely to cause GvHD. Gene expression analysis using Affymetrix arrays was performed on CML, CD34+ (GvL target cells), as well as on epithelial and epidermal cells (GvHD targets). Additional external expression data was also used (e.g. GeneNotes[29]). By subtracting GvHD associated expression from GvL associated expression, a list of 687 SNP-containing genes was generated. The SNPs were then filtered with the requirement that they were validated by genotypic frequency data, and that peptide carriers (homozygous positive + heterozygous positive) were at least 10% of the population and that non-peptide carriers (homozygous negative) were also at least 10% of the population (data provided by HapMap[30]). Finally, those peptides with the highest HLA binding prediction scores, and fulfilling reasonable proteasomal cleavage prediction scores were reported as GvL-relevant peptide candidates. These candidates, listed in Table 4 of Section 9, represent the best targets for the *ex vivo* proliferation of GvL-specific T cells for immunotherapeutic application, according to the criteria of this system.

2.4 Ontology

A look at the schematic of the PeptideCheck database (Appendix A and B) reveals the complexity involved when processing large amounts of biological data. To help manage the complexity of the HLA system, we produced an MHC Ontology (Section 10). An ontology is a collection of defined terms which are joined through defined relations. Ontologies are written in a format which is easily computer-accessible (easily incorporated into computer programs).

In the case of HLA, this is necessary to manage the relationships that different HLA alleles have among each other. The HLA nomenclature is able to do this to a limited extent. In some cases, the genetic similarities among HLA alleles can be inferred by the HLA name alone (see the Introduction of Section 10). However, a formal definition was lacking, and so the MHC Ontology was produced.

The resulting MHC Ontology has an important subdivision, named the HLA Ontology. Because of the rapid rate of expansion of HLA data, it made sense to maintain a relatively stable, upper level ontology (MHC Ontology), which then imports the HLA component from an external Ontology (the HLA Ontology). In this way, every time that the HLA data from the IMGT/HLA Database are updated, the HLA Ontology can be automatically refreshed, and imported into the MHC Ontology. To achieve this automation, a java program was written to create the HLA Ontology dynamically in OWL format. The stable, MHC Ontology, was composed “by hand” using the Protégé ontology editor. The MHC Ontology consists of 106 classes and 7 relations. The HLA Ontology utilizes the relations from the MHC Ontology and consists of 6649 classes, as of

IMGT/HLA release 2.20.0.

The formal, computer-readable definition of HLA allelic hierarchy has allowed for improvement in the interface of the PeptideCheck web resource. The ontology has been incorporated into the Module Explorer section of the PeptideCheck website. This allows the user to more easily access the HLA alleles of interest. Importantly, the problem of A*02 alleles spilling over into the A*92 group has been resolved. Although it is not evident from the nomenclature, A*92 alleles actually belong to the A*02 two-digit group. This is also true for B*15 alleles spilling over into B*95. Because this system is automatically updated with every IMGT/HLA database update, the ontology provides a reliable means of representing and organizing HLA alleles.

3 Conclusions

Managing Polymorphism

The concept of polymorphism poses not only a challenge for HSCT, but also for bioinformatics. The extreme level of variation in the HLA system makes the attempts of nomenclature committees seem futile. Here the MHC and HLA Ontologies have proven helpful. The large number of HLA protein variants makes it impossible to quantify the functional discrepancies for every mismatch situation. The HistoCheck website provides a convenient way to analyses these discrepancies no matter how many new alleles are added to the database. Eluting and sequencing peptide binding data for each HLA variant would require extreme amounts of financial resources. The modular concept of HLA provides a means of applying resources efficiently to maximize the impact of new data for HLA peptide binding prediction.

Outside of the MHC, polymorphisms throughout the genome play a role in HSCT via mHags. Although the volume of data when considering the all SNPs of the genome is several orders of magnitude greater than that for HLA, modern computers have little problem processing millions of entries, as long as programmers pay careful attention to efficiency and optimization. In the case of PeptideCheck, the flexibility offered by the InterSystems Caché databases was very important for achieving such optimizations. It was almost surprising to see that even when considering such quantities of peptide candidates, ranking based upon HLA peptide binding score was able to reproduce validate mHags.

Lessons Learned

Several lessons were learned throughout the course of these investigations. A very large number of external resources were required to create these bioinformatics solutions. Most of these resources could be tapped via automated online systems. In particular, accessing data in XML format via HTTP is a very convenient way for bioinformaticians to utilize the resources they provide for each other. By providing such services, resources such as NCBI or HapMap can increase the efficiency of their contribution to further computer-dependant research.

The contribution of x-ray crystallographic data was indispensable in creating HistoCheck and the modular concept of HLA. This is a difficult and costly procedure, but the benefits for computational analysis justify theses costs. The importance of quality over quantity was underscored by the SNP typing results in Section 9. Here, only validated SNPs from the dbSNP could be reproduced. The correlation between HLA peptide binding prediction scores, and the likelihood of identifying mHags was surprisingly strong (Section 9).

Final Word

In conclusion, computational biology is an important tool for deepening our understanding of concepts in immunogenetics, when it is combined with experimental data and validation. Using computer systems to organize and present biological data is practical and necessary. Using database and algorithms to simulate biological processes and yield new knowledge is more challenging, but possible. The PeptideCheck system shows that at least a significant portion of biological events can be simulated, even if not every peptide target can be found. Some processes are easily represented in computer systems (e.g. finding peptides encoded by missense SNPs), while the high level of complexity makes other processes elusive (e.g. finding peptides encoded by alternate splicing events).

There is a high likelihood that an effective treatment for leukemia will one day result from the efforts in the field of immunogenetics. Computer systems will help.

References

[1] E. Spierings, B. Wieles, and E. Goulmy. Minor histocompatibility antigens–big in tumour therapy. *Trends Immunol*, 25(2):56–60, 2004. 1471-4906 (Print)Journal ArticleResearch Support, Non-U.S. Gov'tResearch Support, U.S. Gov't, P.H.S.Review.

[2] C. Yee. Adoptive t cell therapy: Addressing challenges in cancer immunotherapy. *J Transl Med*, 3(1):17, 2005. 1479-5876 (Electronic)Journal article.

[3] K. L. Knutson, W. Wagner, and M. L. Disis. Adoptive t cell therapy of solid cancers. *Cancer Immunol Immunother*, 55(1):96–103, 2006. 0340-7004 (Print)Journal ArticleResearch Support, N.I.H., ExtramuralReview.

[4] S. A. Rosenberg. Progress in human tumour immunology and immunotherapy. *Nature*, 411(6835):380–4, 2001. 0028-0836 (Print)Journal ArticleReview.

[5] L. Hambach and E. Goulmy. Immunotherapy of cancer through targeting of minor histocompatibility antigens. *Curr Opin Immunol*, 17(2):202–10, 2005. 0952-7915 (Print)Journal ArticleResearch Support, Non-U.S. Gov'tReview.

[6] J. Robinson, M. J. Waller, P. Parham, N. de Groot, R. Bontrop, L. J. Kennedy, P. Stoehr, and S. G. Marsh. Imgt/hla and imgt/mhc: sequence databases for the study of the major histocompatibility complex. *Nucleic Acids Res*, 31(1):311–4, 2003. 1362-4962Journal Article.

[7] E. Goulmy, R. Schipper, J. Pool, E. Blokland, J. H. Falkenburg, J. Vossen, A. Gratwohl, G. B. Vogelsang, H. C. van Houwelingen, and J. J. van Rood. Mismatches of minor histocompatibility antigens between hla-identical donors and recipients and the development of graft-versus-host disease after bone marrow transplantation. *N Engl J Med*, 334(5):281–5, 1996. 0028-4793Journal Article.

[8] M. A. Saper, P. J. Bjorkman, and D. C. Wiley. Refined structure of the human histocompatibility antigen hla-a2 at 2.6 a resolution. *J Mol Biol*,

219(2):277–319, 1991. 0022-2836 (Print)Comparative StudyJournal ArticleResearch Support, Non-U.S. Gov'tResearch Support, U.S. Gov't, P.H.S.

[9] G. Chelvanayagam. A roadmap for hla-a, hla-b, and hla-c peptide binding specificities. *Immunogenetics*, 45(1):15–26, 1996. 0093-7711Journal Article.

[10] M. Bhasin, H. Singh, and G. P. Raghava. Mhcbn: a comprehensive database of mhc binding and non-binding peptides. *Bioinformatics*, 19(5):665–6, 2003. 1367-4803Journal Article.

[11] C. Bade-Doeding, H. A. Elsner, B. Eiz-Vesper, A. Seltsam, U. Holtkamp, and R. Blasczyk. A single amino-acid polymorphism in pocket a of hla-a∗6602 alters the auxiliary anchors compared with hla-a∗6601 ligands. *Immunogenetics*, 56(2):83–8, 2004. 0093-7711Journal Article.

[12] C. Bade-Doeding, B. Eiz-Vesper, C. Figueiredo, A. Seltsam, H. A. Elsner, and R. Blasczyk. Peptide-binding motif of hla-a∗6603. *Immunogenetics*, 56(10):769–72, 2005. 0093-7711Journal Article.

[13] C. Bade-Doeding, D. S. DeLuca, A. Seltsam, R. Blasczyk, and B. Eiz-Vesper. Amino acid 95 causes strong alteration of peptide position pomega in hla-b∗41 variants. *Immunogenetics*, 59(4):253–9, 2007. 0093-7711 (Print)Journal ArticleResearch Support, Non-U.S. Gov't.

[14] H. Rammensee, J. Bachmann, and S. Stevanovic. Mhc ligands and peptide motifs. *Land Bioscience*, Molecular Biology Intelligence Unit, 1997.

[15] K. L. Rock and A. L. Goldberg. Degradation of cell proteins and the generation of mhc class i-presented peptides. *Annu Rev Immunol*, 17:739–79, 1999. 0732-0582 (Print)In VitroJournal ArticleResearch Support, Non-U.S. Gov'tResearch Support, U.S. Gov't, P.H.S.Review.

[16] R. E. Toes, A. K. Nussbaum, S. Degermann, M. Schirle, N. P. Emmerich, M. Kraft, C. Laplace, A. Zwinderman, T. P. Dick, J. Muller, B. Schonfisch, C. Schmid, H. J. Fehling, S. Stevanovic, H. G. Rammensee, and H. Schild. Discrete cleavage motifs of constitutive and immunoproteasomes revealed by quantitative analysis of cleavage products. *J Exp Med*, 194(1):1–12, 2001. 0022-1007 (Print)Journal ArticleResearch Support, Non-U.S. Gov't.

[17] N. P. Emmerich, A. K. Nussbaum, S. Stevanovic, M. Priemer, R. E. Toes, H. G. Rammensee, and H. Schild. The human 26 s and 20 s proteasomes generate overlapping but different sets of peptide fragments from a model protein substrate. *J Biol Chem*, 275(28):21140–8, 2000. 0021-9258 (Print)Journal Article.

[18] I. Ginodi, T. Vider-Shalit, L. Tsaban, and Y. Louzoun. Precise score for the prediction of peptides cleaved by the proteasome. *Bioinformatics*, 24(4):477–83, 2008. 1460-2059 (Electronic)Journal ArticleResearch Support, N.I.H., ExtramuralResearch Support, Non-U.S. Gov't.

[19] C. Kesmir, A. K. Nussbaum, H. Schild, V. Detours, and S. Brunak. Prediction of proteasome cleavage motifs by neural networks. *Protein Eng*, 15(4):287–96, 2002. 0269-2139 (Print)Journal Article.

[20] M. Nielsen, C. Lundegaard, O. Lund, and C. Kesmir. 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. 0093-7711 (Print)Journal ArticleResearch Support, N.I.H., ExtramuralResearch Support, Non-U.S. Gov'tResearch Support, U.S. Gov't, P.H.S.

[21] K. C. Parker, M. A. Bednarek, and J. E. Coligan. Scheme for ranking potential hla-a2 binding peptides based on independent binding of individual peptide side-chains. *J Immunol*, 152(1):163–75, 1994. 0022-1767Journal Article.

[22] H. Rammensee, J. Bachmann, N. P. Emmerich, O. A. Bachor, and S. Stevanovic. Syfpeithi: database for mhc ligands and peptide motifs. *Immunogenetics*, 50(3-4):213–9, 1999. 0093-7711Journal ArticleReviewReview, Tutorial.

[23] P. A. Reche, J. P. Glutting, and E. L. Reinherz. Prediction of mhc class i binding peptides using profile motifs. *Hum Immunol*, 63(9):701–9, 2002. 0198-8859Journal Article.

[24] V. Brusic, G. Rudy, G. Honeyman, J. Hammer, and L. Harrison. Prediction of mhc class ii-binding peptides using an evolutionary algorithm and artificial neural network. *Bioinformatics*, 14(2):121–30, 1998. 1367-4803 (Print)Journal ArticleResearch Support, Non-U.S. Gov't.

[25] S. Buus, S. L. Lauemoller, P. Worning, C. Kesmir, T. Frimurer, S. Corbet, A. Fomsgaard, J. Hilden, A. Holm, and S. Brunak. Sensitive quantitative predictions of peptide-mhc binding by a 'query by committee' artificial neural network approach. *Tissue Antigens*, 62(5):378–84, 2003. 0001-2815 (Print)Journal Article.

[26] I. A. Doytchinova, P. Guan, and D. R. Flower. Epijen: a server for multistep t cell epitope prediction. *BMC Bioinformatics*, 7:131, 2006. 1471-2105 (Electronic)Journal ArticleResearch Support, Non-U.S. Gov't.

[27] M. Nielsen, C. Lundegaard, P. Worning, S. L. Lauemoller, K. Lamberth, S. Buus, S. Brunak, and O. Lund. Reliable prediction of t-cell epitopes using neural networks with novel sequence representations. *Protein Sci*, 12(5):1007–17, 2003. 0961-8368 (Print)Journal Article.

[28] E. Spierings, J. Drabbels, M. Hendriks, J. Pool, M. Spruyt-Gerritse, F. Claas, and E. Goulmy. A uniform genomic minor histocompatibility antigen typing methodology and database designed to facilitate clinical applications. *PLoS ONE*, 1:e42, 2006. 1932-6203 (Electronic)Journal Article.

[29] O. Shmueli, S. Horn-Saban, V. Chalifa-Caspi, M. Shmoish, R. Ophir, H. Benjamin-Rodrig, M. Safran, E. Domany, and D. Lancet. Genenote: whole genome expression profiles in normal human tissues. *C R Biol*, 326(10-11):1067–72, 2003. 1631-0691 (Print)Journal ArticleResearch Support, Non-U.S. Gov't.

[30] The International HapMap Consortium. The international hapmap project. *Nature*, 426(6968):789–96, 2003. 1476-4687 (Electronic)Journal ArticleMulticenter StudyResearch Support, Non-U.S. Gov'tResearch Support, U.S. Gov't, P.H.S.

Part III. Enclosed Articles

4 The Immunoinformatics of Cancer Immunotherapy

Title

The immunoinformatics of cancer immunotherapy

Authors

DeLuca, D.S. and Blasczyk, R.

Published by invitation in

Tissue Antigens, 2007

REVIEW ARTICLE

The immunoinformatics of cancer immunotherapy

D. S. DeLuca & R. Blasczyk

Institute for Transfusion Medicine, Hannover Medical School, Hannover, Germany

Key words
antigen presentation; immunoinformatics; immunotherapy

Correspondence
Dr Rainer Blasczyk
Institute for Transfusion Medicine
Hannover Medical School
Carl-Neuberg-Street 1
30625 Hannover
Germany
Tel: +49 511 532 6700
Fax: +49 511 532 2079
e-mail: blasczyk.rainer@mh-hannover.de

Received 16 July 2007; accepted 16 July 2007

doi: 10.1111/j.1399-0039.2007.00914.x

Abstract

We review here the developments in the field of immunoinformatics and their present and potential applications to the immunotherapeutic treatment of cancer. Antigen presentation plays a central role in the immune response, and as a result in immunotherapeutic methods such as adoptive T-cell transfer and antitumor vaccination. We therefore extensively review the current technologies of antigen presentation prediction, including the next generation predictors, which combine proteasomal processing, transporter associated with antigen processing and major histocompatibility complex (MHC)-binding prediction. Minor histocompatibility antigens are also relevant targets for immunotherapy, and we review the current systems available, SNEP and SiPep. Here, antigen presentation plays a key role, but additional types of data are also incorporated, such as single nucleotide polymorphism data and tissue/cell-type expression data. Current systems are not capable of handling the concept of immunodominance, which is critical to immunotherapy, but efforts have been made to model general aspects of the immune system. Although tough challenges lie ahead, when measuring the field of immunoinformatics on its contributions thus far, one can expect fruitful developments in the future.

Introduction

The cells of the immune system hold great potential to be harnessed for their therapeutic effects against cancer. Potential therapies include adoptive transfer of *ex vivo*-expanded antigen-specific T cells (1–3), as well as *in vivo* vaccination (4). In the case of adoptive transfer, a highly specific reactivity can be achieved using peptide-pulsed or transfected cells, as well as using artificial antigen-presenting cells (5–8). In addition to successful immunotherapeutic applications in mouse tumor models (9), clinical plausibility in humans is continually becoming established (10–14). The identification of the optimal antigens to use either in *ex vivo* T-cell stimulation or as anticancer vaccines is a crucial step in the development of immunotherapies (15, 16). Here, there is major potential for support from the field of computational biology.

Immunoinformatics is emerging as a field with growing significance and application in the clinical setting (17). For example, intense effort has been made to computationally represent the antigen presentation process (18–20). For immunotherapy, antigen targets can be of a tumor-specific nature (21). In this case, experiments are necessary to identify tumor-specific target proteins, at which point antigen presentation prediction can be very useful for identifying immunoreactive domains. Further experimentation by major histocompatibility complex (MHC)-binding assays and cytotoxic T lymphocyte (CTL) assays is then required to select the functionally relevant antigens from the list of theoretical candidates (22).

In the context of hematopoietic stem cell transplantation, minor histocompatibility antigens (mHags) arise from the polymorphic genetic profile of a particular patient with respect to that of a stem cell donor (23). The graft *vs* leukemia (GvL) effect, which occurs when donor lymphocytes eliminate the residual malignant cells of the patient, can be considered the most established form of immunotherapy (24). Exploiting the GvL effect, while minimizing graft *vs* host disease (GvHD), is the major challenge of the hematopoietic stem cell transplantation field. Here, immunoinformatics contributes to the effort to identify GvL-inducing mHags by mining polymorphism data, expression profiles as well as antigen presentation predictions (25, 26).

Antigen presentation

Antigen presentation is the process leading to the presentation of potential T-cell epitopes on the cell surface by MHC. In the case of presentation by MHC class I, this process involves protein degradation, peptide transport into the endoplasmic reticulum via transporter associated with antigen processing (TAP), and MHC binding (27). The solved crystal structures of these components are depicted in Figure 1. Algorithms have been developed to predict each of these fundamental steps in this process, as well as combination algorithms, which integrate the individual types of prediction. An overview of such algorithms is shown in Table 1.

Protein cleavage

The proteasome is the central focus of the efforts of immunoinformatics to describe protein degradation. There are two sources of data, which are used to train cleavage prediction systems. On the one hand, there are peptide sequences generated by the *in vivo* degradation of specific proteins (28–30). These data are extremely insightful into the manner in which the proteasome operates. In particular, it is clear from these digestion assays that, while there are favored cleavage sites, the same protein can be cleaved at different sites. The result is a pool of peptides, which in many cases overlap in the original protein sequence. The limiting aspect of this high-quality data is simply that few proteins have been analyzed in this way and that a larger training set is desirable. This has caused other immunoinformaticians to turn to a more plentiful source of data: naturally presented peptides eluted from MHC proteins. It is assumed that such natural peptides must have been cleaved by the proteasome and can therefore be used in training predictors in combination with the source protein sequences. The inclusion of peptides, which have been eluted from MHC proteins having differing binding motifs, should weaken an MHC-binding bias inherent in this data. However, as we will discuss later, these kinds of systems may still be predicting MHC binding to some extent. The unavoidable consequence of this kind of data, however, is the fact that cleavage sites in proteins will be missed, when the resulting peptide does not bind MHC, and is not included in the eluted peptide dataset as a result.

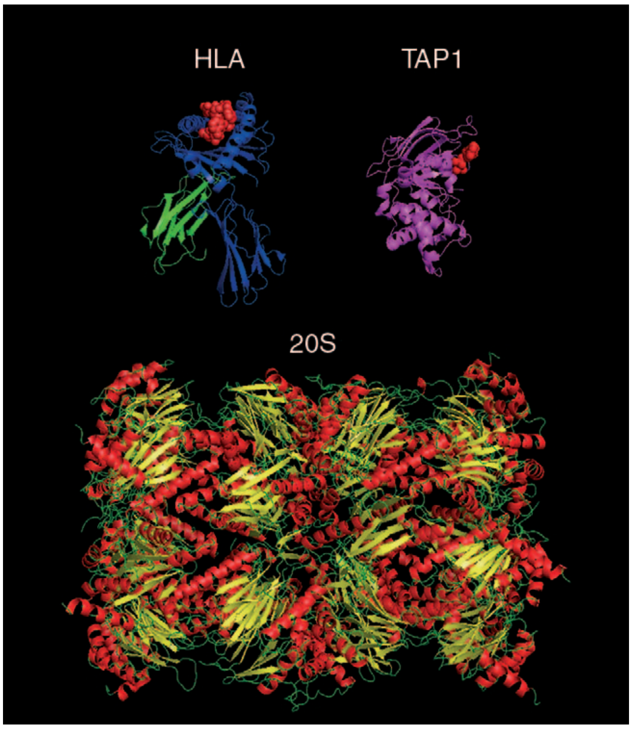


Figure 1 The major players in antigen presentation – human leukocyte antigen (HLA), transporter associated with antigen processing (TAP) and the proteasome. The crystal structures of three proteins central to the antigen presentation process are depicted in their relative sizes to one another. The HLA protein is in complex with β_2 microglobulin in green and a decamer peptide in red. TAP1 is shown in isolation because the details of TAP complex formation are still unknown. The ADP-binding site is shown in red. The barrel-like 20S subunit of the yeast proteasome is colored according to secondary structure. The following PDB files were used: 1HHH (71), 1RYP (72), and 1JJ7 (73).

In addition to the proteasome, aminopeptidases are responsible for shortening the peptides, which are ultimately presented by MHC (31, 32). These enzymes act on the N-terminus of the peptide. Because there is currently no system for predicting aminopeptidase digestion, a certain gray area surrounds the determination of the N-terminus of predicted peptides. Because of this fact, it is arguable that predicting digested protein fragments should concentrate mainly on the C-terminus, with the assumption that for a given C-terminal cleavage site, peptidases will produce an array of peptides with differing N-terminal sites. This is supported by the observation of naturally presented peptides that result from the same C-terminal cleavage site but have differing lengths (33). There is another caveat concerning peptide length that should be mentioned here. The requirement by some prediction systems that a peptide be C-terminally cleaved but not contain internal, 'peptide-destroying' cleavage sites may be too strict. The *in vivo* digestion assays show the existence of peptides stemming from overlapping sequences in the original protein. This strongly suggests that an internal cleavage site does not disqualify a sequence region as a potential peptide. This is further supported by the latest attempts at including the internal cleavage site as a disqualifier, which did not improve proteasomal cleavage prediction performance (20).

Transporter associated with antigen processing

After proteasomal cleavage, peptide sequences are transported into the endoplasmic reticulum by the TAP protein. This process is assisted by various chaperones, such as tapasin, calreticulin and the disulfide isomerase ERp57 (34).

The peptide-binding motif of human TAP has been deciphered by combinatorial libraries. This and other

Table 1 Overview of antigen presentation prediction systems[a]

Resource	Proteasomal processing	TAP binding	HLA binding	Availability
Multistep epitope prediction				
NetCTL 1.2 (20)	NetChop 3.0 (65)	SMM (39)	NetMHC 3.0 (45)	http://www.cbs.dtu.dk/services/NetCTL/
IEDB (66)	SMM (18)	SMM (39)	ANN, ARB, and SMM	http://www.immuneepitope.org/
EpiJen v1.0 (19)	Quantitative matrix (67)	Tap additive model (68)	MHCPred (69)	http://www.jenner.ac.uk/EpiJen/
Minor histocompatibility antigen prediction				
SiPep (25)	None	None	BIMAS (41), SYF (43), nHLAPred (70)	http://www.sipep.org/
SNEP (26)	None	None	SYF (43)	http://elchtools.de/SNEP/

ANN, artificial neural networks; ARB, average relative binding; HLA, human leukocyte antigen; IEDB, immune epitope database and analysis resource; SMM, stabilized matrix method; SYF, SYFPEITHI; TAP, transporter associated with antigen processing.

[a] Numbers in parentheses are literature citations.

studies showed significant amino acid preferences, particularly at the C-terminus and the three N-terminal positions (35, 36). Peptides of up to 16 amino acids are preferred, with lengths of 8–12 binding most efficiently (37). For the purpose of predicting the rate at which TAP transports a particular peptide into the ER, TAP-peptide-binding affinity can be used as an estimator of transport rate, because these two values have been shown to correspond (38). In the pioneering work of Peters et al., weight matrices were fitted to TAP-peptide affinity data (39). Phenylalanine, tyrosine, and arginine were shown to be particularly favored at the C-terminus. We will discuss the implications of this effort in the section on combined predictions below.

Putting it together with MHC-binding prediction

The final step in antigen presentation is a classical subject of immunoinformatics: MHC binding. This is the most restrictive step involved in antigen presentation. It is estimated that only 1 out of 200 peptides will bind a given MHC class I allele with sufficient strength to elicit a CTL response (40). Many approaches have been taken to predict MHC–peptide binding (41–48), and it is useful at this point to discuss the prediction of all steps together, which are needed for comprehensive prediction of the entire presentation pathway.

The means of evaluating multistep antigen presentation prediction is to use a dataset of naturally processed MHC-bound peptides in combination with the original peptide sequences (the SYFPEITHI database being the accepted gold standard for curated naturally processed peptides). In the work of Tenzer et al., the stabilized matrix method (SMM) for TAP transport was combined with a novel SMM-based cleavage prediction and MHC-binding prediction algorithm (18). The novel cleavage prediction method, dubbed ProteaSMM, showed improvements over older methods such as paproc and NetChop 2.0. Furthermore, a significant improvement was demonstrated when combining prediction steps. In particular, the combined prediction showed marked improvement over proteasomal processing prediction alone. However, a 'breakthrough' improvement over MHC-binding prediction alone was not observed. This is also true for NetCTL, the artificial neural network-based attempt at multistep antigen presentation prediction, in which only a minimal improvement over MHC-binding prediction alone was shown (20). It is difficult to make conclusions here about how this reflects the biology of these processes, or whether this is a result of training or testing artifacts. On the one hand, MHC-binding predictors could have hidden proteasomal cleavage and TAP transports element to them if these motifs are found directly in the presented peptides. On the other hand, MHC binding is considered the bottleneck in the presentation pathway, and one would therefore expect that the respective prediction step would weigh the most heavily.

Vaccination

The appeal of antitumor vaccination is the fact that it takes advantage of *in vivo* processes and has the potential to harness the full power of the immune system, in contrast to the more artificial *ex vivo* expansion of T cells. The use of synthetic peptide vaccines enables one to achieve a high level of specificity, with relative ease of production (49). It has been shown that the peptides predicted to bind MHC can elicit a tumor-killing CTL response (50). An effective methodology for determining new tumor-specific peptide epitopes involves the application of antigen prediction algorithms to a tumor-associated protein, then experimental confirmation of the MHC-binding affinity, and finally stimulation of CTLs with peptide-loaded dendritic cells (51).

Class II MHC-bound epitopes play an important role in the antitumor response by the activation of CD4+ T cells and help maintain effective CTL response (52, 53). Historically, predicting peptide binding of class II MHC has been much more challenging than that of class I (54–56).

The reason for this lies in the fact that class II proteins bind to peptides of variable length and that the core anchor residues cannot be readily identified. Application of artificial neural networks has been a successful endeavor, at least for predicting the high-affinity side of the peptide binder spectrum (54). A review of class II predictions has shown that such neural network-based predictions are effective when sufficient peptide-binding data are available but that motif-based versions are favored for small peptide datasets (56). Clearly, prediction for class II alleles needs to be improved and expanded to include more allelic variants, considering the central role that it plays in the immune response. This holds true not only for antitumor vaccination but presumably for the immune response to mHags as well (23).

mHags prediction

mHags are peptides, which are presented by MHC on the cell surface of an individual, and cause an alloreactive immune response as a result of their absence in the individual from whom the attacking lymphocytes originate. A database of known minors is hosted by the Leiden University Medical Center and can be accessed online at http://www.lumc.nl/5033/dbminor/. In the context of a hematopoietic stem cell transplantation, such antigens can cause GvHD (57) as well as the therapeutic GvL effect (24). Finding target antigens that minimize the former and maximize the latter is the goal of immunoinformatic efforts, which combine antigen presentation with additional biological data including polymorphism and gene expression data (25, 26).

Polymorphism data are necessary to identify gene variants, which could result in the differential expression of peptides. The majority of known ‘minors’ result from a single nucleotide polymorphism (SNP) in a coding region of a gene, resulting in an amino acid substitution. However, insertions, deletions, frameshift mutations, mutations resulting in stop codons, mutations eliminating stop codons, splice site and promoter mutations and whole gene deletions (58) are all conceivable. Most of these kinds of mutations are computable with conventional means using databases such as the dbMHC. What will remain difficult in the immediate future is the identification of peptide variations that result from mutations outside of the gene encoding the peptide (e.g. mutations that disable activator proteins). The SiPep web service, which aims to predict mHags, used the dbMHC as well as data from the HapMap project to compute variant peptides (25). In this system, coding nonsynonymous mutations are considered. Because most known minors do result from this type of polymorphism, it is not unreasonable to concentrate solely on them. Future applications, however, would do well to consider the other types of relevant mutations. Another mHag prediction system is SNEP (25), which also focuses on coding nonsynonymous SNPs. In this system, the CONFLICT and VARIANT annotations in SWISS-PROT provide the polymorphism data.

For leukemia treatment, if mHags are to be used to expand T cell for adoptive transfer, they should induce GvL without GvHD. This would be the case when the minor is either restricted to the malignant cells or restricted to the cells of the hematopoietic origin. This latter case is acceptable because the patient’s blood system will be replaced by the graft. Indeed, hematopoietic expression is the criteria recommend by the authors of SiPep. In this system, the types of tissue in which the peptide candidates are expressed are obtained from Stanford University’s SOURCE database (http://hrweb.stanford.edu/source/). SNEP does not address the issue of cell- or tissue-specific gene expression. Additionally, while SNEP is limited to the MHC-binding step of antigen presentation, SiPep includes proteasomal processing. Another advantage of SiPep is the inclusion of SNP frequency data. Our own unpublished investigation of SNP frequencies from dbSNP entries resulted in the following: of an 100 patients typed for nonvalidated SNPs, no SNP could be confirmed; for SNPs that were validated with frequencies from the HapMap project, the majority were confirmed. This strongly underscores the importance of such frequency data. The SNEP system however relies on SWISS-PROT, and not on dbSNP, so the polymorphism data it used could be of a higher quality than nonvalidated dbSNP data, in particular for the VARIANT entries that are validated.

Immunodominance – the next frontier

Despite the great advances in the prediction algorithms mentioned above, the understanding of immunodominance is still a distant, elusive goal. The ability to predict TAP binding is an important milestone because peptide presentation has been shown to vary greatly dependent on TAP affinities (59). There are of course additional factors that influence the peptide landscape found on the cell surface, many of which are probably still unknown. For example, before proteasomal processing, the expression levels and turnover rates of proteins could influence the final concentrations in which they are ultimately presented on the cell surface. The utilization of existing protein turnover prediction programs should be considered (60). Furthermore, it has been shown that the cleavage specificities vary between the constitutive proteasome and the immunoproteasome (28). Clearly, the induction of immunoproteasome expression by cytokines could influence immunodominance. *In vitro* assays for proteasomal digestion and human leukocyte antigen binding have shown poor correlation to the respective *in silico* predictions from the late 1990s (61). This underscores a need for further assays to evaluate the latest generation of prediction algorithms. To make things more complicated, there is evidence that the dominant peptides presented in a given MHC protein change depending on which additional MHC proteins an individual carries (62).

A path that could take us closer to understanding immunodominance is *in silico* modeling of the immune system. Biological processes, including the immune system, can be modeled using differential equations (63) or agent-based models (64). Differential equations can be used to mathematically describe the concentrations, number or levels of biological entities with respect to each other and with respect to time. While such models are very elegant, they are quite difficult to implement correctly and are very inflexible. For these reasons, there is a tendency toward agent-based systems of modeling the immune system. Here, individual entities are represented in the computer, given coordinates in space, and behaviors are defined, which describe how the entities interface with the immediate surrounding environment. Each time step is then calculated iteratively, and the progress of the system can be observed. Currently, such immune system simulations have found more applications outside of the field of immunology than inside (64). This is the innovative field of artificial immune systems, in which immune system-inspired computer algorithms are applied to 'real-world' engineering applications. This field is based on a dialogue between immunologists and computer scientists, and although the computer scientists are benefiting more than the immunologists from this field at the moment, there is good hope that further advances in immune system simulations will be useful in generating scientific knowledge.

Concluding remarks

The immunoinformatic community has made an important contribution to the effort to develop T-cell-based cancer therapies by providing comprehensive antigen presentation systems. Breakthroughs in the biological model of antigen presentation will have to be met with further experimental data-driven bioinformatics. The hurdle of reliable class II peptide binding prediction is a high priority, considering its significance in the immune system. The quality of mHag prediction systems has the potential to improve as more and more of the entries in the underlying databases become validated. Whether immune system modeling can take the quantum leap to predict immunodominance is unknown. Perhaps, the vision of agent-based representations of peptide sets, competing for access to the antigen presentation machinery, will become a reality. This, together with the extension of the model to include T cell receptor (TCR) interaction and signaling downstream of the TCR, could provide useful insights into this complicated biological process.

References

1. Yee C. Adoptive T cell therapy: addressing challenges in cancer immunotherapy. *J Transl Med* 2005: **3**: 17.
2. Knutson KL, Wagner W, Disis ML. Adoptive T cell therapy of solid cancers. *Cancer Immunol Immunother* 2006: **55**: 96–103.
3. Rosenberg SA. Progress in human tumour immunology and immunotherapy. *Nature* 2001: **411**: 380–4.
4. Ribas A, Butterfield LH, Glaspy JA, Economou JS. Current developments in cancer vaccines and cellular immunotherapy. *J Clin Oncol* 2003: **21**: 2415–32.
5. Maus MV, Thomas AK, Leonard DG et al. Ex vivo expansion of polyclonal and antigen-specific cytotoxic T lymphocytes by artificial APCs expressing ligands for the T-cell receptor, CD28 and 4-1BB. *Nat Biotechnol* 2002: **20**: 143–8.
6. Oelke M, Maus MV, Didiano D, June CH, Mackensen A, Schneck JP. Ex vivo induction and expansion of antigen-specific cytotoxic T cells by HLA-Ig-coated artificial antigen-presenting cells. *Nat Med* 2003: **9**: 619–24.
7. Yee C, Gilbert MJ, Riddell SR et al. Isolation of tyrosinase-specific CD8+ and CD4+ T cell clones from the peripheral blood of melanoma patients following in vitro stimulation with recombinant vaccinia virus. *J Immunol* 1996: **157**: 4079–86.
8. Yee C, Thompson JA, Byrd D et al. Adoptive T cell therapy using antigen-specific CD8+ T cell clones for the treatment of patients with metastatic melanoma: in vivo persistence, migration, and antitumor effect of transferred T cells. *Proc Natl Acad Sci U S A* 2002: **99**: 16168–73.
9. Melief CJ, Toes RE, Medema JP, van der Burg SH, Ossendorp F, Offringa R. Strategies for immunotherapy of cancer. *Adv Immunol* 2000: **75**: 235–82.
10. Hsu FJ, Benike C, Fagnoni F et al. Vaccination of patients with B-cell lymphoma using autologous antigen-pulsed dendritic cells. *Nat Med* 1996: **2**: 52–8.
11. Nestle FO, Alijagic S, Gilliet M et al. Vaccination of melanoma patients with peptide- or tumor lysate-pulsed dendritic cells. *Nat Med* 1998: **4**: 328–32.
12. Marchand M, van Baren N, Weynants P et al. Tumor regressions observed in patients with metastatic melanoma treated with an antigenic peptide encoded by gene MAGE-3 and presented by HLA-A1. *Int J Cancer* 1999: **80**: 219–30.
13. Thurner B, Haendle I, Roder C et al. Vaccination with mage-3A1 peptide-pulsed mature, monocyte-derived dendritic cells expands specific cytotoxic T cells and induces regression of some metastases in advanced stage IV melanoma. *J Exp Med* 1999: **190**: 1669–78.
14. Kolb HJ, Holler E. Adoptive immunotherapy with donor lymphocyte transfusions. *Curr Opin Oncol* 1997: **9**: 139–45.
15. Offringa R, van der Burg SH, Ossendorp F, Toes RE, Melief CJ. Design and evaluation of antigen-specific vaccination strategies against cancer. *Curr Opin Immunol* 2000: **12**: 576–82.
16. Rosenberg SA. A new era for cancer immunotherapy based on the genes that encode cancer antigens. *Immunity* 1999: **10**: 281–7.
17. Brusic V, Petrovsky N. Immunoinformatics – the new kid in town. *Novartis Found Symp* 2003: **254**: 3–13; discussion 13–22, 98–101, 250–2.

18. Tenzer S, Peters B, Bulik S et al. Modeling the MHC class I pathway by combining predictions of proteasomal cleavage, TAP transport and MHC class I binding. *Cell Mol Life Sci* 2005: **62**: 1025–37.
19. Doytchinova IA, Guan P, Flower DR. EpiJen: a server for multistep T cell epitope prediction. *BMC Bioinformatics* 2006: **7**: 131.
20. Larsen MV, Lundegaard C, Lamberth K et al. An integrative approach to CTL epitope prediction: a combined algorithm integrating MHC class I binding, TAP transport efficiency, and proteasomal cleavage predictions. *Eur J Immunol* 2005: **35**: 2295–303.
21. Van den Eynde BJ, van der Bruggen P. T cell defined tumor antigens. *Curr Opin Immunol* 1997: **9**: 684–93.
22. Del Val M, Schlicht HJ, Ruppert T, Reddehase MJ, Koszinowski UH. Efficient processing of an antigenic sequence for presentation by MHC class I molecules depends on its neighboring residues in the protein. *Cell* 1991: **66**: 1145–53.
23. Hambach L, Goulmy E. Immunotherapy of cancer through targeting of minor histocompatibility antigens. *Curr Opin Immunol* 2005: **17**: 202–10.
24. Barrett AJ, van Rhee F. Graft-versus-leukaemia. *Baillieres Clin Haematol* 1997: **10**: 337–55.
25. Halling-Brown M, Quartey-Papafio R, Travers PJ, Moss DS. SiPep: a system for the prediction of tissue-specific minor histocompatibility antigens. *Int J Immunogenet* 2006: **33**: 289–95.
26. Schuler MM, Donnes P, Nastke MD, Kohlbacher O, Rammensee HG, Stevanovic S. SNEP: sNP-derived epitope prediction program for minor H antigens. *Immunogenetics* 2005: **57**: 816–20.
27. Pamer E, Cresswell P. Mechanisms of MHC class I –restricted antigen processing. *Annu Rev Immunol* 1998: **16**: 323–58.
28. Emmerich NP, Nussbaum AK, Stevanovic S et al. The human 26 S and 20 S proteasomes generate overlapping but different sets of peptide fragments from a model protein substrate. *J Biol Chem* 2000: **275**: 21140–8.
29. Toes RE, Nussbaum AK, Degermann S et al. Discrete cleavage motifs of constitutive and immunoproteasomes revealed by quantitative analysis of cleavage products. *J Exp Med* 2001: **194**: 1–12.
30. Tenzer S, Stoltze L, Schonfisch B et al. Quantitative analysis of prion-protein degradation by constitutive and immuno-20S proteasomes indicates differences correlated with disease susceptibility. *J Immunol* 2004: **172**: 1083–91.
31. Craiu A, Akopian T, Goldberg A, Rock KL. Two distinct proteolytic processes in the generation of a major histocompatibility complex class I-presented peptide. *Proc Natl Acad Sci U S A* 1997: **94**: 10850–5.
32. Serwold T, Gaw S, Shastri N. ER aminopeptidases generate a unique pool of peptides for MHC class I molecules. *Nat Immunol* 2001: **2**: 644–51.
33. Barnea E, Beer I, Patoka R et al. Analysis of endogenous peptides bound by soluble MHC class I molecules: a novel approach for identifying tumor-specific antigens. *Eur J Immunol* 2002: **32**: 213–22.
34. Abele R, Tampe R. The ABCs of immunology: structure and function of TAP, the transporter associated with antigen processing. *Physiology (Bethesda)* 2004: **19**: 216–24.
35. van Endert PM, Riganelli D, Greco G et al. The peptide-binding motif for the human transporter associated with antigen processing. *J Exp Med* 1995: **182**: 1883–95.
36. Uebel S, Kraas W, Kienle S, Wiesmuller KH, Jung G, Tampe R. Recognition principle of the TAP transporter disclosed by combinatorial peptide libraries. *Proc Natl Acad Sci U S A* 1997: **94**: 8976–81.
37. Scholz C, Tampe R. The intracellular antigen transport machinery TAP in adaptive immunity and virus escape mechanisms. *J Bioenerg Biomembr* 2005: **37**: 509–15.
38. Gubler B, Daniel S, Armandola EA, Hammer J, Caillat-Zucman S, van Endert PM. Substrate selection by transporters associated with antigen processing occurs during peptide binding to TAP. *Mol Immunol* 1998: **35**: 427–33.
39. Peters B, Bulik S, Tampe R, Van Endert PM, Holzhutter HG. Identifying MHC class I epitopes by predicting the TAP transport efficiency of epitope precursors. *J Immunol* 2003: **171**: 1741–9.
40. Yewdell JW, Bennink JR. Immunodominance in major histocompatibility complex class I-restricted T lymphocyte responses. *Annu Rev Immunol* 1999: **17**: 51–88.
41. Parker KC, Bednarek MA, Coligan JE. Scheme for ranking potential HLA-A2 binding peptides based on independent binding of individual peptide side-chains. *J Immunol* 1994: **152**: 163–75.
42. Rognan. Molecular dynamics simulation of MHC-peptide complexes as a tool for predicting potential T cell epitopes. *Biochemistry* 1994: **33**: 11476–86.
43. Rammensee H, Bachmann J, Emmerich NP, Bachor OA, Stevanovic S. SYFPEITHI: database for MHC ligands and peptide motifs. *Immunogenetics* 1999: **50**: 213–9.
44. Rognan D, Lauemoller SL, Holm A, Buus S, Tschinke V. Predicting binding affinities of protein ligands from three-dimensional models: application to peptide binding to class I major histocompatibility proteins. *J Med Chem* 1999: **42**: 4650–8.
45. Buus S, Lauemoller SL, Worning P et al. Sensitive quantitative predictions of peptide-MHC binding by a 'Query by Committee' artificial neural network approach. *Tissue Antigens* 2003: **62**: 378–84.
46. Nielsen M, Lundegaard C, Worning P et al. Reliable prediction of T-cell epitopes using neural networks with novel sequence representations. *Protein Sci* 2003: **12**: 1007–17.
47. Reche PA, Glutting JP, Zhang H, Reinherz EL. Enhancement to the RANKPEP resource for the prediction of peptide binding to MHC molecules using profiles. *Immunogenetics* 2004: **56**: 405–19.
48. DeLuca DS, Khattab B, Blasczyk R. A modular concept of HLA for comprehensive peptide binding prediction. *Immunogenetics* 2007: **59**: 25–35.
49. Disis ML, Cheever MA. HER-2/neu oncogenic protein: issues in vaccine development. *Crit Rev Immunol* 1998: **18**: 37–45.
50. Lu J, Celis E. Use of two predictive algorithms of the world wide web for the identification of tumor-reactive T-cell epitopes. *Cancer Res* 2000: **60**: 5223–7.

51. Rongcun Y, Salazar-Onfray F, Charo J et al. Identification of new HER2/neu-derived peptide epitopes that can elicit specific CTL against autologous and allogeneic carcinomas and melanomas. *J Immunol* 1999: **163**: 1037–44.
52. Hung K, Hayashi R, Lafond-Walker A, Lowenstein C, Pardoll D, Levitsky H. The central role of CD4(+) T cells in the antitumor immune response. *J Exp Med* 1998: **188**: 2357–68.
53. Kalams SA, Walker BD. The critical need for CD4 help in maintaining effective cytotoxic T lymphocyte responses. *J Exp Med* 1998: **188**: 2199–204.
54. Brusic V, Rudy G, Honeyman G, Hammer J, Harrison L. Prediction of MHC class II-binding peptides using an evolutionary algorithm and artificial neural network. *Bioinformatics* 1998: **14**: 121–30.
55. Marshall KW, Wilson KJ, Liang J, Woods A, Zaller D, Rothbard JB. Prediction of peptide affinity to HLA DRB1*0401. *J Immunol* 1995: **154**: 5927–33.
56. Yu K, Petrovsky N, Schonbach C, Koh JY, Brusic V. Methods for prediction of peptide binding to MHC molecules: a comparative study. *Mol Med* 2002: **8**: 137–48.
57. Goulmy E, Schipper R, Pool J et al. Mismatches of minor histocompatibility antigens between HLA-identical donors and recipients and the development of graft-versus-host disease after bone marrow transplantation. *N Engl J Med* 1996: **334**: 281–5.
58. Murata M, Warren EH, Riddell SR. A human minor histocompatibility antigen resulting from differential expression due to a gene deletion. *J Exp Med* 2003: **197**: 1279–89.
59. Fruci D, Lauvau G, Saveanu L et al. Quantifying recruitment of cytosolic peptides for HLA class I presentation: impact of TAP transport. *J Immunol* 2003: **170**: 2977–84.
60. Bachmair A, Finley D, Varshavsky A. In vivo half-life of a protein is a function of its amino-terminal residue. *Science* 1986: **234**: 179–86.
61. Kessler JH, Beekman NJ, Bres-Vloemans SA et al. Efficient identification of novel HLA-A(*)0201-presented cytotoxic T lymphocyte epitopes in the widely expressed tumor antigen PRAME by proteasome-mediated digestion analysis. *J Exp Med* 2001: **193**: 73–88.
62. Paston SJ, Dodi IA, Madrigal JA. Progress made towards the development of a CMV peptide vaccine. *Hum Immunol* 2004: **65**: 544–9.
63. Chan CC, Stark J, George AJ. Analysis of cytokine dynamics in corneal allograft rejection. *Proc Biol Sci* 1999: **266**: 2217–23.
64. Forrest S, Beauchemin C. Computer immunology. *Immunol Rev* 2007: **216**: 176–97.
65. Nielsen M, Lundegaard C, Lund O, Kesmir C. The role of the proteasome in generating cytotoxic T-cell epitopes: insights obtained from improved predictions of proteasomal cleavage. *Immunogenetics* 2005: **57**: 33–41.
66. Peters B, Sidney J, Bourne P et al. The immune epitope database and analysis resource: from vision to blueprint. *PLoS Biol* 2005: **3**: e91.
67. Doytchinova IA, Flower DR. Class I T-cell epitope prediction: improvements using a combination of proteasome cleavage, TAP affinity, and MHC binding. *Mol Immunol* 2006: **43**: 2037–44.
68. Doytchinova I, Hemsley S, Flower DR. Transporter associated with antigen processing preselection of peptides binding to the MHC: a bioinformatic evaluation. *J Immunol* 2004: **173**: 6813–9.
69. Guan P, Hattotuwagama CK, Doytchinova IA, Flower DR. MHCPred 2.0: an updated quantitative T-cell epitope prediction server. *Appl Bioinformatics* 2006: **5**: 55–61.
70. Bhasin M, Raghava GP. A hybrid approach for predicting promiscuous MHC class I restricted T cell epitopes. *J Biosci* 2007: **32**: 31–42.
71. Madden DR, Garboczi DN, Wiley DC. The antigenic identity of peptide-MHC complexes: a comparison of the conformations of five viral peptides presented by HLA-A2. *Cell* 1993: **75**: 693–708.
72. Groll M, Ditzel L, Lowe J et al. Structure of 20S proteasome from yeast at 2.4 A resolution. *Nature* 1997: **386**: 463–71.
73. Gaudet R, Wiley DC. Structure of the ABC ATPase domain of human TAP1, the transporter associated with antigen processing. *EMBO J* 2001: **20**: 4964–72.

5 HistoCheck: Rating of HLA Class I and II Mismatches

Title

HistoCheck: rating of HLA class I and II mismatches by an internet-based software tool

Authors

Elsner, H.A., DeLuca, D., Strub, J. and Blasczyk, R.

Published in

Bone Marrow Transplantation, 2004

Because permission was not given to reproduce this article here, please see the original publication: Elsner, H.A., DeLuca, D., Strub, J. and Blasczyk, R. (2004) HistoCheck: rating of HLA class I and II mismatches by an internet-based software tool, Bone Marrow Transplant, 33, 165-169.

6 HistoCheck: Evaluating MHC Similarities.

Title

HistoCheck: Evaluating Structural and Functional MHC Similarities.

Authors

DeLuca, D.S. and Blasczyk, R.

Published by invitation in

Immunoinformatics (Methods in Molecular Biology), 2007

30

HistoCheck

Evaluating Structural and Functional MHC Similarities

David S. DeLuca and Rainer Blasczyk

Summary

The HistoCheck webtool provides clinicians and researchers with a way of visualizing and understanding the structural differences among related major histocompatibility complex (MHC) molecules. In the clinical setting, human leukocyte antigen (HLA) matching of hematopoietic stem cell donors and recipients is essential to minimize "graft versus host disease" (GvHD). Because exact HLA matching is often not possible, it is important to understand which alleles present the same structures (HLA–peptide complexes) to the T-cell receptor (TCR) despite having different amino acid sequences. HistoCheck provides a summary of amino acid mismatches, positions, and functions as well as 3-dimensional (3D) visualizations. In this chapter, we describe how HistoCheck is used and offer advice in interpreting the query results

Key Words: Histocheck; HLA; MHC; class I; class II; peptide; binding; GvHD; donor; stem cell transplantation; matching; T-cell receptor

1. Introduction

The collection of genes known as the major histocompatibility complex (MHC) was discovered during studies initiated by J. Dausset, R. Payne and J. J. van Rood, which attempted to describe a genetically inherited system of alloantigens (antigens resulting from genetic discrepancies during transplantation) in the 1950s ***(1–3)***. During the early 1960s, multi-transfused patients and parous women were shown to often have circulating antibodies against alloantigens, now known to be encoded by the human form of MHC—human leukocyte antigen (HLA). Consequently, anti-HLA antibody screening is a standard practice when matching organ donors and recipients. Later, it became

From: *Methods in Molecular Biology, vol. 409: Immunoinformatics: Predicting Immunogenicity In Silico*
Edited by: D. R. Flower © Humana Press Inc., Totowa, NJ

clear that MHC-derived proteins restrict the specificity of the antigen receptor expressed on the surface of T lymphocytes and thus play a major role in the regulation of the immune response ***(4)***.

In the context of organ transplantation between non–HLA-identical donors and recipients, the recipient's T cells identify the donor's HLA proteins as foreign and initialize an immune response against the transplant. Consequently, the survival rate among recipients of HLA-matched organs is significantly higher than when mismatches are present ***(5,6)***.

HLA matching for organ donors and recipients is complicated by HLA's high rate of polymorphism. The latest release of the IMGT/HLA database contains 2,088 alleles ***(7)***. Exact matching across multiple HLA loci (e.g., HLA-A, HLA-B, HLA-C, and HLA-DRB1) is very difficult. For kidney, heart, cornea, and pancreas transplantations, "low-resolution" matching is used—HLA alleles are only required to belong to the same serological group. For hematopoietic stem cell transplantations during leukemia therapy, "high-resolution" matching is required; patient and recipient alleles are required to produce the same protein sequence. After total body irradiation for eliminating malignant hematopoietic cells, leukemia patients need to receive a new hematopoietic and immune system through stem cell transplantation. From the perspective of the donor's immune cells, the recipient's entire body is foreign, which leads to the so-called *graft versus host disease* (GvHD).

The likelihood of finding a high-resolution match for stem cell transplantation is low, and therefore, clinicians often seek a "next-best" match. This requires an understanding of which amino acid differences are not expected to result in a functional change to the HLA protein. Here, the selective binding of HLA to short peptide sequences, as well as the T-cell receptor (TCR), is of the greatest interest. Amino acid differences in regions of the protein that do not play a role in peptide or TCR binding could be acceptable between stem cell donor and recipient.

The peptide binding groove is encoded by exons 2 and 3 for class I HLA and exon 2 for class II HLA. The binding groove is formed by a beta-sheet "floor" with two alpha-helical "walls." Peptides bind by squeezing in between the alpha helices, typically deeply anchored at the second amino acid from the N terminus, as well as the C-terminal position. The TCR contacts the binding groove from above, interacting with the surface amino acids of the alpha helices and peptide ***(8)***.

HistoCheck (http://www.histocheck.org) is an online tool which helps clinicians and researchers visualize the amino acid substitutions of HLA alleles so that they can make informed judgments about their functional similarity.

HistoCheck provides crystallography-based 3-dimensional (3D) visualizations of the allelic mismatches by highlighting amino acids substitution positions. The user is provided with dissimilarity scores (DSSs) for the amino acids involved as well as an over-all DSS for the two alleles *(9)*.

2. Implementation

HistoCheck is written in Java, runs on a Tomcat application server, utilizes servlets, Java server pages, and a MySQL database. The HLA alleles and their sequences are updated regularly via the IMGT/HLA database: ftp://ftp.ebi.ac.uk/pub/databased/imgy/mhc/hla/ *(7)*.

2.1. Three-Dimensional Visualization

GIF images of the HLA structures with highlighted mismatches are generated on a linux server using RasMol version 2.7.1.1. A description of RasMol script commands can be found in the University of Massachusetts web server http://www.umass.edu/microbio/rasmol/distrib/rasman.htm. Chime can be integrated into the HTML of a website using the EMBED tag. Here is an example:

```
<embed src="PDB_FILE_NAME.pdb" bgcolor=black display3d=
cartoon color3d=chain height="590" width ="600" startspin="false"
script="script SCRIPT_NAME.spt;">
```

Commands used in the *.spt file correspond largely with standard RasMol commands.

2.2. The DSS Algorithm

In addition to providing information on the specific amino acid substitutions involved between two HLA alleles, HistoCheck generates a DSS, which attempts to quantify the overall functional differences between the two alleles (*see* **Note 1**) *(10)*. The score is based on the Risler substitution matrix as well as data on the function of specific amino acids positions (i.e., their role in peptide binding or TCR interaction) (*see* **Note 2**) *(11)*. The score is generated by

1. summing the Risler scores across all mismatches,
2. dividing this score by 100,
3. adding a penalty of 1 for each mismatch that occurs on a position that either interacts with the TCR or the peptide, or both.

An example calculation is given in Table 1.

Table 1
Calculating the dissimilarity score for A*2402 and A*2304

Position	Mismatch	Function	Penalty	Risler score
144	Lysine → Glutamine	–		13
151	Histidine → Arginine	TCR	+1	64
156	Glutamine → Leucine	PEP	+1	27
166	Aspartic acid → Glutamic acid	TCR	+1	30
167	Glycine → Tryptophan	TCR+PEP	+1	87
		Total	4	221
		Divide Risler scores by 100		221/100 = 2.21
		Dissimilarity score		4 + 2.21 = 6.21

PEP, Peptide contact site; TCR, T-cell receptor.

The dissimilarity score is based on the Risler scores of mismatched amino acids combined with penalties for positions which interact with the TCR or peptide. Note that although position 157 is involved in both TCR contact and peptide binding, the penalty is only counted once.

3. Application

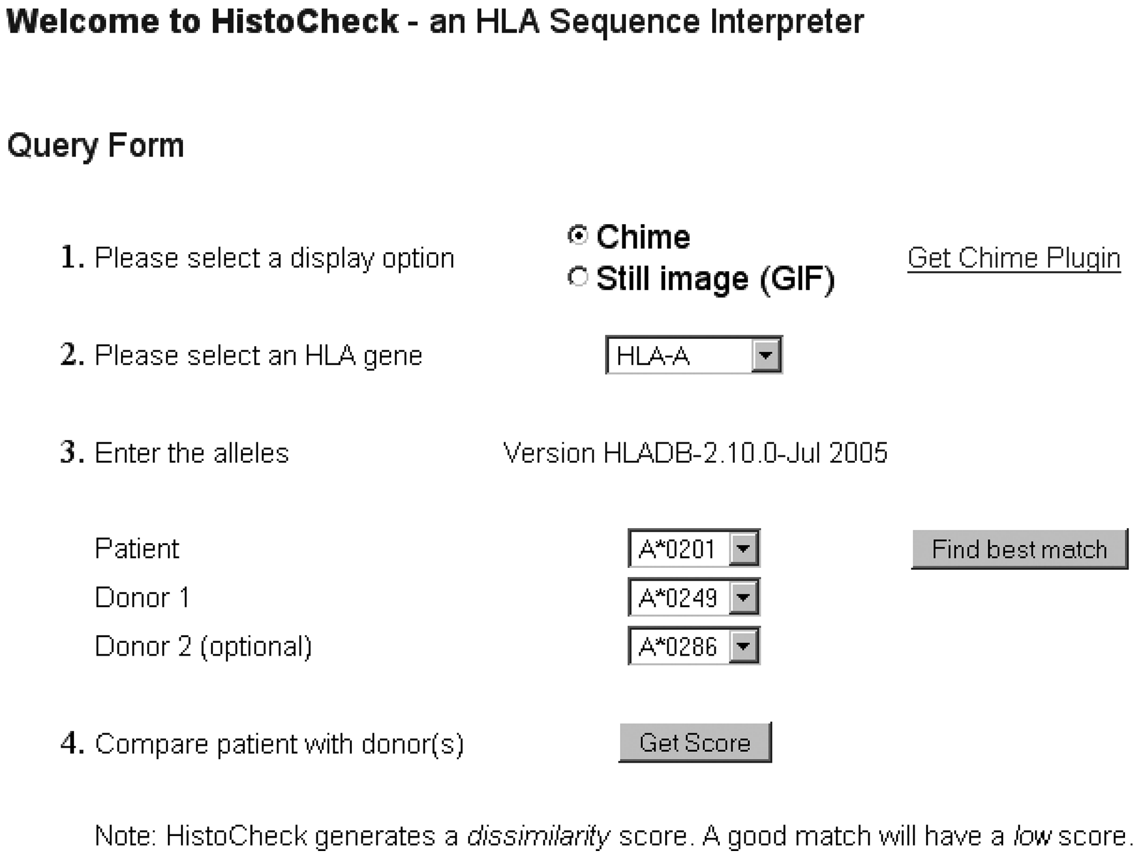
HistoCheck can be accessed online at http://histocheck.org using any javascript-enabled browser. Although HistoCheck is available free of charge, first-time users are required to register for a user name and password, because the developers are interested in what kinds of medical and research institutes find HistoCheck userful.

3.1. Comparing a Patient's HLA to Specific Donor HLA

After signing in to HistoCheck, the user is presented with a query form (Fig. 1). The first option is the type of display to be used in showing the 3D structure of HLA. Chime is a web-browser plug-in that presents molecules interactively in 3D, allowing the user to rotate the molecule and choose between a variety of display options. Alternatively, a still GIF image can be generated, which shows the alleles' 3D structure, but is not interactive.

Next, the user may select one of the following HLA loci: A, B, Cw, DRB1, DRB3, DRB4, DRB5, DQA1, DQB1, DPA1, and DPB1. The specific alleles for donor and recipient can then be specified. Two donors may be specified, for a side-by-side comparison.

The resulting webpage shows a list of amino acid mismatches between donor and recipient (Fig. 2). For each mismatch, the domain, exon, pocket, and amino acid position are displayed (*see* **Note 3**). To help understand the significance of each mismatch, additional information is given: the position's



Welcome to HistoCheck - an HLA Sequence Interpreter

Query Form

1. Please select a display option ● **Chime** ○ **Still image (GIF)**

2. Please select an HLA gene HLA-A

3. Enter the alleles Version HLADB-2.10.0-Jul 2005

Patient A*0201

Donor 1 A*0249

Donor 2 (optional) A*0286

4. Compare patient with donor(s)

Note: HistoCheck generates a *dissimilarity* score. A good match will have a *low* score.

Fig. 1. The query page for the HistoCheck website. The user may choose display options and human leukocyte antigen (HLA) alleles for structural comparison. Patient alleles can be compared directly with donor candidates with the “Get Score” button. Alternatively, all the alleles of a locus can be ranked by similarity to the patient’s allele by clicking the “Find best match” button.

role in binding the peptide and/or TCR, as well as the Risler score for the two amino acids involved (*see* **Note 4**). The combination of functional significance of the position (TCR binding/peptide binding), and the extent of biophysical dissimilarity between the amino acids, is the basis for the DSS (*see* **Note 5**). The summary table lists the total number of mismatches, the affected pockets, total number of mismatches that affect peptide binding, the total number of mismatched positions that interact with the TCR, and the overall DSS.

Underneath the mismatch tables, the HLA mismatches are displayed visually either as a GIF image or in an interactive Chime window. The mismatched positions are highlighted yellow. For class I HLA, the structure is based up HLA-A*0201 in complex with a decameric peptide from Hepatitis B nucleocapsid protein. The α_1, α_2, and α_3 domains are displayed in blue. The α_1 and α_2 domains form the peptide binding groove, which also interacts with the TCR.

The β_2-microglobulin domain is shown in green. A decamer peptide is shown bound to the protein in red. If class II alleles were selected, the 3D structure is based on crystallographic data from HLA-DRA with HLA-DRB1*0101. The β_1 and β_2 domains of DRB1 are shown in dark blue. The α_1 and α_2 domains of DRA are shown in turquoise. The bound 13-mer peptide is shown in green. The α_1 and β_1 domains form the peptide binding groove. Although class II HLA proteins are heterodimers, the user selects only one gene at a time, for simplicity. In this case, only the mismatches for the protein of the selected gene are displayed. Because HLA-DRA, encoding for the alpha chain of the various DR heterodimers, is not polymorphic, it is not offered in the list of genes.

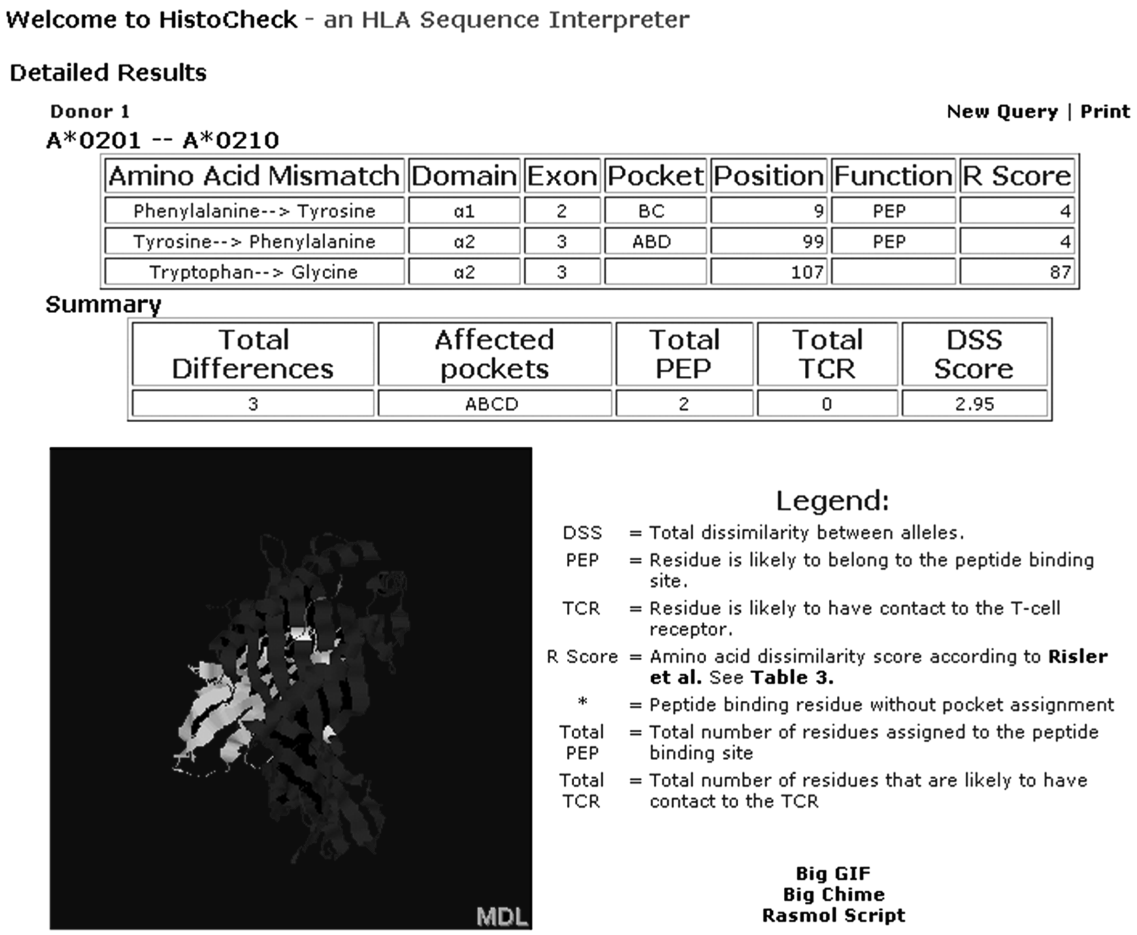
If the Chime display option was selected, the user can rotate the molecule and zoom in on particularly interesting locations. Chime also provides various display options. The default option is "cartoons," which allows one to quickly orient and locate secondary, tertiary, and quaternary structures. Other options, such as wireframe, ball and stick, and space-fill can be used for more detail, once the major landmarks have been identified.

A large GIF image or Chime representation can be obtained by clicking the "Big GIF" or "Big Chime" links. The "RasMol Script" link provides an rsm file, which contains the atomic coordinate information from the standard pdb format, as well as commands which orient the HLA molecule and highlight the mismatches. The rsm files can be downloaded and viewed locally using the RasMol viewer, RasTop 2.0.

3.2. Ranking Alleles by their Similarity to a Patient's HLA

HistoCheck can also be used to find the most similar variants of an allele. The procedure is almost identical to that described in Section 3.1. However, after selecting the donor's allele on the query page, the user may also click the "Find Best Match" button instead of the "Get Score" button. In this case, all of the alleles of the given locus are considered and ranked by ascending DSS (i.e., the most similar alleles are at the top of the list). The ordered list of alleles appears in the right frame, and the mismatch result page for the best match is displayed in the center frame.

For example, if HLA-A*0201 is chosen as the donor's allele, a report comparing A*0201 with A*0209 appears in the center frame. Because A*0201 and A*0209 have no amino acid differences in the key domains (α_1 and α_2), the DSS is zero. These alleles are different at position 236 of the mature protein, but this position is part of the α_3 domain, which does not interact with the TCR or peptide. Although no mismatches are reported, the footnote "Additional differences found outside key domains" as well as the 3D image with the highlighted



Welcome to HistoCheck - an HLA Sequence Interpreter

Detailed Results

Donor 1

New Query | Print

A*0201 -- A*0210

Amino Acid Mismatch	Domain	Exon	Pocket	Position	Function	R Score
Phenylalanine--> Tyrosine	α1	2	BC	9	PEP	4
Tyrosine--> Phenylalanine	α2	3	ABD	99	PEP	4
Tryptophan--> Glycine	α2	3		107		87

Summary

Total Differences	Affected pockets	Total PEP	Total TCR	DSS Score
3	ABCD	2	0	2.95

Legend:

DSS	= Total dissimilarity between alleles.
PEP	= Residue is likely to belong to the peptide binding site.
TCR	= Residue is likely to have contact to the T-cell receptor.
R Score	= Amino acid dissimilarity score according to **Risler et al.** See **Table 3.**
*	= Peptide binding residue without pocket assignment
Total PEP	= Total number of residues assigned to the peptide binding site
Total TCR	= Total number of residues that are likely to have contact to the TCR

Big GIF
Big Chime
Rasmol Script

Fig. 2. The results page from a HistoCheck query. Here, the user has chosen to compare HLA-A*0201 with A*0210. Three amino acid differences were found at positions 9, 99, and 107. Positions 9 and 99 are involved in peptide binding. The SSM score quantifies the functional differences of these alleles. In the crystallographic structure of HLA bound to a peptide, the three mismatch positions are highlighted. Two mismatches can be seen on the beta-sheet, and one in a loop structure on the lower right.

mismatch appears. In the ranking of the most similar alleles to A*0201 on the right, one can see that A*0201 has a zero mismatch score with A*0209, A*0266, and A*0275. Clicking on the allele's name in this list brings up the detailed report for the comparison. Clicking on the fourth allele in the list, A*0268, one can see a single amino acid substitution: arginine to lysine. at position 157. Although this position is in the α_2 domain, it does not interact directly with the peptide or the TCR and is therefore of low significance. Visual inspection of the 3D structure shows that position 157 is part of the domain's alpha helix, but faces away from the peptide binding groove. Furthermore, arginine and lysine (both long and basic) are structurally very similar, as reflected by the very low Risler score (3). It can be concluded that despite a mismatch in the α_2 domain, A*0201 and A*0268 can be expected to bind the same peptides and appear identical to the TCR.

3.3. Interpretation of DSS

As described in Section 2.2, the DSS is based up the functional role of the mismatched positions, as well as the structural similarity of the amino acids involved. The example involving A*0201 mentioned above describes comparisons where it is quite clear that the amino acid differences are unlikely to affect HLA function. The best matches are of course those with DSS of zero, indicating that there are no differences in the key domains. Amino acid substitutions which are in the key domains, but which are not involved in peptide binding or contact with the TCR, are likely to be tolerable. Mismatches in peptide or TCR-binding regions could only be expected to be tolerable when the Risler score is very low (below 10). See (*see* **Notes 1–3**) for more information on interpreting the DSS.

3.4. Chime Installation

Interactive protein viewers are useful tools for understanding protein structure. Chime is a web-browser plug-in, allowing for integration into websites. Chime works with Internet Explorer, Netscape, and FireFox. Downloading Chime requires free registration at the MDL website. Good instructions on downloading and installing Chime can be found at the University of Massachusetts website http://www.umass.edu/microbio/chime/ getchime.htm..

Although the Chime installation is straightforward for all versions of Internet Explorer, problems may arise when installing for the newest Netscape and FireFox browsers. A trick for installing chime in these browsers is worth mentioning here. The instructions given below refer to MDL Chime version 6.2 SP6.

1. Install Chime normally for Internet Explorer.
2. Copy the npchime.dll file from the Internet Explorer plug-in folder (C:\Program Files\Internet Explorer\plugins\).
3. Paste the file into the plug-in folder of FireFox or Netscape. For FireFox the folder is likely to be C:\Program Files\Mozilla Firefox\plugins\.

Acknowledgments

We are grateful for the ongoing contributions of Holger-Andreas Elsner to the HistoCheck project. The development of HistoCheck was partially funded by the German José Carreras Leukemia-Foundation (grant DJCLS R 04/01).

Notes

1. This manuscript describes the functionality of HistoCheck at end of 2005. The next version of HistoCheck will involve several improvements. New crystallographic data are available, which have been re-analyzed to determine the functional roles of HLA amino acid positions. This analysis includes locus-specific definitions for TCR and peptide interactions. Furthermore, static correlations between certain HLA mismatches and GvHD have been identified. These "special mismatches" will be highlighted in HistoCheck's mismatch report, and the reference papers will be sighted.
2. Alternatives to the current DSS will be offered. The BLOSUM62 scoring matrix, for example, has delivered improvements in the area of sequence alignments. Whether this matrix is better than the Risler matrix for comparing HLA alleles has not been determined. This question is complicated by the fact that such matrices are based on the assumption that the rate of amino acid substitution among related proteins is proportional to amino acid similarity. The HLA binding groove is an exception to this rule because of the evolutionary pressure for diversity, driven by the need to respond to rapidly mutating pathogens. For this reason, a dissimilarity algorithm will be provided, which weighs the HLA positions according to the variability analysis provided by Reche et al. ***(13)***.
3. A refreshing aspect of HistoCheck in the age of black-box-bioinformatics (i.e., artificial neural networks and hidden Markov models) is that the primary biological data are provided to the user. These so-called "hard data" include the nucleic acid and protein sequences that have been validated by numerous work groups and are, in effect, irrefutable. The mismatched positions reported by HistoCheck are primary data, and the user is left with the freedom to interpret them. Other aspects of HistoCheck can be considered secondary data (also called "soft" or "semi-soft" data). The crystallographically determined structures of HLA are models, whose limitations should be recognized. In particular, the fluidity and elasticity of protein structures are not represented in these models. It can be expected that the conformation of loops, for example, differs greatly in aqueous versus crystal environments. That said, comparison of many crystallographic HLA structures shows that the protein backbone is remarkably conserved. Although "semi-soft," crystallographic models are extremely informative, concerning tertiary/quaternary protein structure, using this data to draw conclusions about TCR interactions and peptide binding can be considered secondary or even tertiary data.
4. Risler's similarity scores are also soft data. The scores are based on the rate of amino acid substitution among structurally similar proteins. HistoCheck's DSS is an attempt to summarize secondary data concerning amino acid substitutions. That this score is highly theoretical and removed from primary data is indisputable. In a preliminary analysis performed with more than 1,700 HLA class I mismatched transplant pairs from the hematopoietic stem cell transplant component of the 13th International Histocompatibility Workshop (Effie Petersdorf, Fred Hutchinson

Cancer Research Center, Seattle, WA), the DSS was not superior in predicting the severity of GvHD compared to just counting the number of HLA class I mismatches (unpublished data). Furthermore, a small preliminary study did not show a correlation between the DSS and T-cell alloreactivity in vitro ***(12)***. Because this study was performed in an allogeneic transplantation setting, in which non-HLA differences (i.e., minor histocompatibility antigens) affected alloreactivity, it is unclear to which extent non-HLA differences overshadowed HLA similarities. To clarify this point, further studies involving autologous cells, modified to express additional HLA proteins, are necessary.

5. HistoCheck's DSS is an elementary mathematical model that represents a first step in quantifying the structural differences between HLA alleles. HistoCheck users are encouraged to study the primary data that this website provides, such as number and location of amino acid substitutions, and to examine the 3D structures provided in order to make informed conclusions about the similarity/dissimilarity of HLA alleles.

References

1. Dausset, J. (1954). Leuco-agglutinins IV. Leuco-agglutinins and blood transfusion. *Vox Sang* 4, 190–8.
2. Payne, R. & Rolfs, M. R. (1958). Fetomaternal leukocyte incompatibility. *J Clin Invest* 37, 1756–63.
3. Van Rood, J., Eernisse, J. G. & van Leeuwen, A. (1958). Leukocyte antibodies in sera from pregnant women. *Nature* 181, 1735–6.
4. Zinkernagel, R. M. & Doherty, P. C. (1974). Restriction of in vitro T cell-mediated cytotoxicity in lymphocytic choriomeningitis within a syngeneic or semiallogeneic system. *Nature* 248, 701–2.
5. Saba, N. & Flaig, T. (2002). Bone marrow transplantation for nonmalignant diseases. *J Hematother Stem Cell Res* 11, 377–87.
6. Hansen, J. A., Gooley, T. A., Martin, P. J., Appelbaum, F., Chauncey, T. R., Clift, R. A., Petersdorf, E. W., Radich, J., Sanders, J. E., Storb, R. F., Sullivan, K. M. & Anasetti, C. (1998). Bone marrow transplants from unrelated donors for patients with chronic myeloid leukemia. *N Engl J Med* 338, 962–8.
7. Robinson, J., Waller, M. J., Parham, P., de Groot, N., Bontrop, R., Kennedy, L. J., Stoehr, P. & Marsh, S. G. (2003). IMGT/HLA and IMGT/MHC: sequence databases for the study of the major histocompatibility complex. *Nucleic Acids Res* 31, 311–4.
8. Saper, M. A., Bjorkman, P. J. & Wiley, D. C. (1991). Refined structure of the human histocompatibility antigen HLA-A2 at 2.6 A resolution. *J Mol Biol* 219, 277–319.
9. Elsner, H. A., DeLuca, D., Strub, J. & Blasczyk, R. (2004). HistoCheck: rating of HLA class I and II mismatches by an internet-based software tool. *Bone Marrow Transplant* 33, 165–9.

10. Elsner, H. A. & Blasczyk, R. (2002). Sequence similarity matching: proposal of a structure-based rating system for bone marrow transplantation. *Eur J Immunogenet* 29, 229–36.
11. Risler, J. L., Delorme, M. O., Delacroix, H. & Henaut, A. (1988). Amino acid substitutions in structurally related proteins. A pattern recognition approach. Determination of a new and efficient scoring matrix. *J Mol Biol* 204, 1019–29.
12. Heemskerk, M. B., Doxiadis, I. I., Roelen, D. L., Claas, F. H. & Oudshoorn, M. (2005). Letter: the HistoCheck algorithm does not predict T-cell alloreactivity in vitro. *Bone Marrow Transplant* [Epub ahead of print, Sep. 5], with 36, 927–8.
13. Reche, P. A. & Reinherz, E. L. (2003). Sequence variability analysis of human class I and class II MHC molecules: functional and structural correlates of amino acid polymorphisms. *J Mol Biol* 331, 623–41.

7 A Modular Concept of HLA

Title

A modular concept of HLA for comprehensive peptide binding prediction

Authors

DeLuca, D.S., Khattab, B. and Blasczyk, R.

Published in

Immunogenetics, 2007

Immunogenetics
DOI 10.1007/s00251-006-0176-4

ORIGINAL PAPER

A modular concept of HLA for comprehensive peptide binding prediction

David S. DeLuca · Barbara Khattab · Rainer Blasczyk

Received: 14 July 2006 / Accepted: 25 October 2006

Abstract A variety of algorithms have been successful in predicting human leukocyte antigen (HLA)-peptide binding for HLA variants for which plentiful experimental binding data exist. Although predicting binding for only the most common HLA variants may provide sufficient population coverage for vaccine design, successful prediction for as many HLA variants as possible is necessary to understand the immune response in transplantation and immunotherapy. However, the high cost of obtaining peptide binding data limits the acquisition of binding data. Therefore, a prediction algorithm, which applies the binding information from well-studied HLA variants to HLA variants, for which no peptide data exist, is necessary. To this end, a modular concept of class I HLA-peptide binding prediction was developed. Accurate predictions were made for several alleles without using experimental peptide binding data specific to those alleles. We include a comparison of module-based prediction and supertype-based prediction. The modular concept increased the number of predictable alleles from 15 to 75 of HLA-A and 12 to 36 of HLA-B proteins. Under the modular concept, binding data of certain HLA alleles can make prediction possible for numerous additional alleles. We report here a ranking of HLA alleles, which have been identified to be the most informative. Modular peptide binding prediction is freely available to researchers on the web at http://www.peptidecheck.org.

Electronic supplementary material Supplementary material is available in the online version of this article at http://dx.doi.org/10.1007/s00251-006-0176-4 and is accessible for authorized users.

D. S. DeLuca · B. Khattab · R. Blasczyk (✉)
Institute for Transfusion Medicine, Hanover Medical School,
Carl-Neuberg-Str 1,
30625 Hanover, Germany
e-mail: blasczyk.rainer@mh-hannover.de

Keywords Histocompatibility Antigens class I · Variation (genetics)/immunology

Abbreviations

MHCBN	major histocompatibility complex binding database
A_{ROC}	area under the receiver operating characteristic curve
SE	sensitivity
SP	specificity
TP	true positive
TN	true negative
FP	false positive
FN	false negative
P1, P2, ..., P9	portions of the HLA binding groove responsible for binding positions 1, 2, ..., 9 of the peptide

Introduction

The process of peptide presentation on the cell surface is central to the specificity of the immune response. Understanding peptide binding by the human leukocyte antigen (HLA) and its presentation to the T cell receptor is essential in the areas of peptide-based vaccination (Rothbard 1992) and immunotherapy, where the concepts graft versus host disease (Goulmy et al. 1996) and graft versus leukemia play a major role. (Hambach and Goulmy 2005).

Many different kinds of algorithms have been developed or adapted to predict which peptide sequences will bind HLA proteins. The algorithms most often associated with MHC–peptide binding prediction are matrix/motif-based (Parker et al. 1994; Rammensee et al. 1999; Reche et al. 2004), hidden Markov models (Noguchi et al. 2002), and artificial neural networks (Buus et al. 2003; Nielsen et al. 2003). Matrix- and motif-based predictions rely on scores for the 20 amino acids at each position in the peptide. Hidden Markov models are capable of considering sequential dependencies among the amino acids in the peptides. Artificial neural networks are a form of nonlinear regression capable of finding patterns in the peptides that affect binding. These algorithms rely on large amounts of experimental data, i.e. example peptide sequences that have been proven to bind to certain HLA binding grooves. However, there are more than 1,700 distinct HLA proteins (Robinson et al. 2003). HLA polymorphism leads in a varying degree to different specificity for peptide sequences. Because of the high costs of obtaining peptide binding data, peptides have only been determined for a handful of the HLA variants. For these few, well-studied variants, conventional algorithms are capable of accurately predicting peptide binding. However, future applications in adoptive immuno- and cell therapy require that peptide binding to *all* HLA variants be understood to develop patient-specific treatments. The application of this peptide-specific T cell approach is the focus of many research groups, for example those involved in the AlloStem project financed by the European Union.

It is possible to predict binding without the use of peptide data. Molecular dynamics simulations can model peptide binding by calculating the forces exerted on every atom. Unfortunately, this technique is prohibitively computationally intensive, requiring months of processor time to simulate few picoseconds of interaction (Rognan et al. 1994). The latest molecular dynamic techniques require only 8 h of computation, with mediocre accuracy (Davies et al. 2003). However, with thousands of HLA variants and an endless supply of peptide sequences, a much faster prediction is necessary. Several groups have approached this problem using machine learning techniques (Yanover and Hertz 2005; Zhu et al. 2006). These studies have shown an improvement in predictive accuracy when pooling peptides that bind within an HLA supertype, as defined by Sette and Sidney (1999). In this study, we attempt to pool peptide binding data not by grouping whole alleles (into supertypes) but by grouping structural subunits of the HLA molecule that have the same amino acid composition (modules). We also compare this approach to supertype-based pooling.

Modular concept of HLA

The aim of this work is to develop fast and accurate predictions for as many HLA variants as possible by developing a modular concept of HLA. Although HLA polymorphism can be caused by point mutation, it is mainly a result of recombination (Kotsch and Blasczyk 2000). Therefore, although a specific HLA is unique, it may be identical to a second HLA in one region and identical to a third HLA in another region. In this study, we explore the possibility of breaking down HLA into modules and correlating these modules with available peptide binding data. In this way, peptide binding data specific for a small number of HLA variants can be applied to an expanded number of variants. Evidence of the effectiveness of this approach has been previously demonstrated for A*6601, 6602, and 6603 (Bade-Doeding et al. 2004, 2005) and for HLA-DR (Sturniolo et al. 1999).

The part of HLA's peptide binding groove that interacts with a specific position in the bound peptide is known as a pocket. Analysis of crystallographic data in class I HLA has provided definitions of which positions in HLA are responsible for binding certain positions in the peptide (Chelvanayagam 1996). Because of the side chain orientation in the protein's three-dimensional structure, the positions responsible for peptide binding are not sequential. For example, the particular residues in HLA that interact with the N-terminal amino acid (P1=peptide position 1) in the peptide are at positions 5, 7, 33, 59, 62, 63, 66, 99, 159, 163, 167, and 171 (Chelvanayagam 1996). These positions are used to define a module. A module is the sequence of amino acids found at these positions in a specific HLA allele. For a 9-mer peptide, a given allele will have nine modules (P1, P2, ..., P9). This is based upon the nine pockets defined by Chelvanayagam. A similar approach based upon the six specificity pockets (A–F) would also be possible but is not examined here. Because of similarities among HLA alleles, different HLAs can share modules when they possess the same amino acids at the defined positions (Table 1). In this work, the modular concept is applied strictly to class I HLA alleles and 9-mer peptides.

The purpose for developing a modular model of the HLA binding groove is to expand the number of alleles for which peptide binding prediction is possible. We report here an expansion of predictable HLA alleles by a factor of five. To achieve the goal of binding prediction across all HLA alleles, more peptide binding data must be gathered. The question—Which alleles should be studied to further populate the peptide binding database?—can be answered in the context of the modular concept of HLA. Because modules can be shared among many or few alleles, it follows that peptide binding data for certain alleles would contribute more to the number of predictable alleles than others. In this work, we therefore also report a list of alleles that should be studied to efficiently contribute to comprehensive peptide binding prediction.

Table 1 Modules for HLA-B*5302

		Known peptides
P1 pocket	1, 5, 7, 33, 59, 62, 63, 66, 99, 159, 163, 167, 171	150
Module	MYFYRNIYYLWH	
Alleles	B*1537, B*3521, 24, B*3932, B*5101, 4, 6–9, 12–14, 17–20, 22, 24, 26, 28–30, 32, 33, 35, 37, 38, B*5302, 6, B*5605, 6, B*7801–3	
P2 pocket	2, 7, 9, 24, 25, 26, 34, 35, 36, 45, 62, 63, 66, 67, 70, 99, 159, 163, 167	445
Module	YYAVGVRFTRNIFNYYLW	
Alleles	B*3501–9, 11, 12, 14, 17, 18, 21, 22, 24, 27, 29, 30, 32, 34, 36–39, 41–44, 51, 52, 54–58, 61, B*5101, 2, 4–6, 8, 9, 12–15, 17–20, 24, 26, 28–30, 32, 33, 35, 37, 38, B*5301–6, 8, 10, B*7801, 2, 4	
P3 pocket	3, 7, 9, 62, 66, 70, 97, 99, 114, 152, 155, 156, 159, 163	277
Module	YYRINRYDVQLYL	
Alleles	B*1505, 20, 31, 91, B*3501, 3, 7, 10, 13, 19, 20, 24–29, 32, 34, 36, 39, 41, 42, 46, 47, 49, 52, 54–57, B*4403, 7, 13, 26, 29, 30, 36–40, B*4802, B*5301–5, 9, 10	
P4 pocket	4, 62, 65, 66, 69, 70, 155, 156, 159	708
Module	RQITNQLY	
Alleles	B*0813, 25, B*1301–4, 6, 9–13, B*1401, 2, 5, 6, B*1502, 3, 5, 6, 9, 10, 13, 18, 20, 21, 23, 25, 29, 31, 36, 37, 39, 40, 42, 44, 48, 52, 55, 61, 62, 64, 69, 72, 80, 86, 88–91, 93, 98, B*1801–12, 14, 15, 18, 20, B*2712, 16, 18, 23, 29, B*3501–7, 9–13, 15–17, 19–37, 39, 41, 42, 46–52, 54–58, 60, B*3702, B*3801–11, B*3901–7, 9, 10, 12–17, 19, 20, 22–24, 26–32, 34, B*4001–14, 18–21, 24–28, 30, 31, 33–40, 42–61, B*4403, 7, 10, 13, 26, 29–31, 36–40, B*4701–5, B*4801–4, 6, 7, 9–13, B*4901–4, B*5001, 2, 4, B*5101–4, 6, 7, 9, 10, 12–19, 21–24, 26, 28, 30–35, 37, 38, B*5201–8, B*5301–10, B*5518, B*5901, B*7801–5, B*9503	
P5 pocket	5, 69, 70, 73, 74, 97, 114, 116, 152, 155, 156, 159	340
Module	TNTYRDSVQLY	
Alleles	B*1310, B*1505, 20, 31, 52, 91, B*1801, 4–12, 18, 20, B*3501, 7, 10, 15, 19, 20, 23–28, 32, 35, 41, 42, 46–50, 52, 54, 57, B*3907, B*4020, 52, 59, 60, B*4802, B*5301–3, 5, 9, 10	
P6 pocket	6, 7, 9, 22, 24, 66, 69, 70, 73, 74, 97, 99, 114, 116, 133, 147, 152, 155, 156	261
Module	YYFAITNTYRYDSWWVQL	
Alleles	B*1505, 20, 31, B*1804, B*3501, 7, 10, 15, 19, 20, 24, 26–28, 32, 35, 41, 42, 46, 47, 49, 52, 54, 57, B*5301–3, 5, 9, 10	
P7 pocket	7, 73, 77, 97, 114, 116, 133, 146, 147, 150, 152, 155, 156	82
Module	TNRDSWKWAVQL	
Alleles	B*1310, B*1809, B*3527, B*5301, 2, 9, 10, B*5801, 4, 9, 11, Cw*0203	
P8 pocket	8, 73, 76, 77, 80, 97, 143, 146, 147	126
Module	TENIRTKW	
Alleles	A*2414, 52, B*1513, 16, 17, 23, 24, 67, 95, B*2730, B*3801, 5–7, 9–11, B*4406, 18, 25, B*4901, 3, 4, B*5104, 6, B*5301, 2, 4, 6–8, 10, B*5705, B*5801, 4, 9, 11	
P9 pocket	9, 70, 73, 74, 76, 77, 80, 81, 84, 95, 96, 97, 114, 116, 123, 124, 142, 143, 146, 147	75
Module	NTYENIAYIQRDSYIITKW	
Alleles	B*1513, B*5301, 2, 6, 8, 10	

HLA-B*5302 is an example of an allele for which peptide binding prediction is possible by using peptide binding data from related alleles. The nine modules here are lists of amino acids from B*5302 that play a role in binding a particular position in a nonamer peptide. The rows “P[1–9] pocket” contain the definitions of which amino acids positions are responsible for binding the respective position in the peptide according to Chelvanayagam. Each module from B*5302 occurs in other alleles as well. These alleles are listed in the rows designated “Alleles.” The numbers of peptides that are associated with each module are listed in the “Known peptides” column.

Materials and methods

Peptides

HLA peptide binding data were provided by the major histocompatibility complex binding (MHCBN) database (Bhasin et al. 2003). This database attempts to combine peptide binding data from a variety of sources, covering a variety of isolation and affinity-determination methods. It not only includes naturally presented peptides eluted from MHC class I molecules as found in the SYFPEITHI database (Rammensee et al. 1999) but also manually selected peptides that were used for the purpose of testing the ability of a specific sequence to bind HLA, e.g. for analyzing viral escape (Gotch et al. 1988). The binding abilities are summarized into four categories: strong, moderate, weak, and nonbinders. Strong, moderate, and weak binders were all considered to be binders for this work. Nonamers were exclusively used in this work. The peptides, their sequences, allele restriction, and source can be publicly accessed on the Internet (http://www.imtech.res.in/raghava/mhcbn/). Although this database does contain nonbinders, it does not provide enough nonbinding nonamers for testing across many alleles. Therefore, random sequences of peptides were generated and assumed to be nonbinders (Supplementary Table 4). This assumption will be true for the vast majority of sequences, because less than 1% of possible peptide sequences are thought to bind HLA class I (Yewdell and Bennink 1999). The use of random nonbinders has several precedents (Donnes and Elofsson 2002; Reche et al. 2004). Random nonamers were generated by randomly choosing human proteins from the Entrez protein database. Segments of nine amino acids were then randomly chosen from the proteins.

Predictive performance

Predictive performance was calculated using the area under the receiver operating characteristic curve (A_{ROC}). The ROC curve is based upon the prediction's *sensitivity*:

$$\mathrm{SE} = \mathrm{TP}/(\mathrm{TP} + \mathrm{FN})$$

and *specificity*:

$$\mathrm{SP} = \mathrm{TN}/(\mathrm{TN} + \mathrm{FP})$$

where TP=true positives: correctly predicted binders; FN=false negatives: binders incorrectly predicted to be nonbinders; TN=true negatives: correctly predicted nonbinders; FP=false positives: nonbinders incorrectly predicted to bind. The ROC curve is a plot of SE versus 1-SP over a range of thresholds. Performance was only tested when 15 or more peptides were available for training. Peptides used in testing were excluded from the matrix scores by the "take one out" technique. Before performing the prediction for a given peptide, the peptide and all peptides with only one amino acid difference were removed from the training data, and the matrices were calculated without these peptides. To test the modular concept, a "no self" evaluation was done. In this case, the values in the modular matrix were generated and tested for a given allele, without using peptide binding data for that allele. For example, predictions were made for A*0201 using binding data from other alleles (A*0202–0206, 0209, 0211, 0214, 0207, 2603, 6601, 6802, 6901) but excluding peptides proven to bind A*0201. Supertype-based prediction was evaluated similarly: The peptides data of all alleles of a supertype—as defined by Sette and Sidney (1999)—were pooled together, excluding the peptide binding data for the allele in question, and matrices were generated as described for the control matrix (see Matrices and prediction below).

Modules

For our purposes, a *pocket* is the list of positions in HLA which are responsible for binding a particular amino acid position in the peptide. In this study, the pockets were defined per Chevanajagam's analysis of crystallographic HLA data (Chelvanayagam 1996). A *module* is the sequence of amino acids found at the pocket positions for a given allele. Modules were generated by combining the pocket definitions provided by Chevanajagam with the HLA protein sequences available in the IMGT/HLA (International Immunogenetics Information System) database, version 2.10.0 (Table 1; Chelvanayagam 1996; Robinson et al. 2003). Although many related alleles produce the same module sequences, only unique sequences were stored in the database table. A second database table was used to correlate the module sequences with the alleles that posses them.

Matrices and prediction

Two kinds of peptide binding prediction were performed: standard (control) matrix and modular matrix. Both of these matrices are 9×20 and contain values for each amino acid at each position in the nonamers peptide. The following pseudocode demonstrates how the matrix values were generated:

Matrix

```
For each allele
 Retrieve all peptides that bind this allele
 For each peptide binder
  For each position in the peptide
   Count the number of occurrences of each amino acid
 Divide all the scores by the number of peptides for this
 allele
```

Modular matrix

```
For each module
 Retrieve all alleles that have this module
 For all alleles with this module
  Retrieve all binders
  For each binder
   Count the amino acid at the position corresponding to
   this module
 Divide the scores by the number of binders found for
 this module
```

A score for a peptide's binding ability is generated by multiplying the nine corresponding values from the matrix. This score is indicative of the likelihood that this peptide is a binder and can be compared to a threshold to predict binding. The values in the control matrix are the frequencies of the amino acids at the particular positions among binding peptides. In the modular matrix, the values are based upon the frequencies of the amino acids, among binding peptides, specific to a particular module. Because different alleles can have certain modules in common, the module-specific values are based upon peptides that bind to all the alleles that have such module.

Most informative alleles

To determine which alleles would provide the most new modular information when their binders are purified and sequenced, three kinds of ranking were performed. For the *maximum module occurrence* analysis, a score was made for each allele by considering the modules it contains, for which no peptide binding data are available, and counting the number of occurrences of each module amongst all alleles. The *maximum anchor occurrence* analysis was performed the same way, but only the anchor positions 2 and 9 were considered. For the purpose of the *maximum predictable alleles* analysis, a predictable allele was defined as having more than five peptides available for its modules at both anchor positions 2 and 9. To do this, first, the total number of predictable alleles was calculated. Then each given allele was assumed to have peptide binding data, and the number of predictable alleles was recalculated. The difference between the new number and the original number was used as the ranking score.

Results

Module generation

A total of 2,525 modules were created for 1,098 class I HLA alleles. This represents only 29% of the theoretically possible number of modules if all alleles were to have nine unique modules. Conversely, it can be said that 71% of class I HLA sequences are conserved on a modular basis. The number of different modules at each peptide position varies and is dependent on the number of amino acids in contact with the peptide as well as the rate of polymorphism at those positions (Fig. 1). For example, pockets P4 and P8, which do not tightly bind the peptide, produced only 72 and 82 modules respectively compared to 458 for P6.

Partial or complete modular matrices could be generated for all class I HLA proteins. Unfortunately, the majority of these matrices are incomplete. Of the 1,098 HLA class I proteins, 342 matrices that had at least one peptide for each of the nine modules were created. The modular matrix for A*0201 is shown in Table 2 as an example. Most of the alleles that contribute peptide binding data to the A*0201 matrix come from other A*02 alleles. However A*2603, 6601, and 6802 also share a module at P8 with A*0201. The fact that A*0209 shares all nine modules with A*0201 comes as no surprise, because these two alleles are identical in the α1 and α2 domains, which are responsible for peptide binding and T cell interaction.

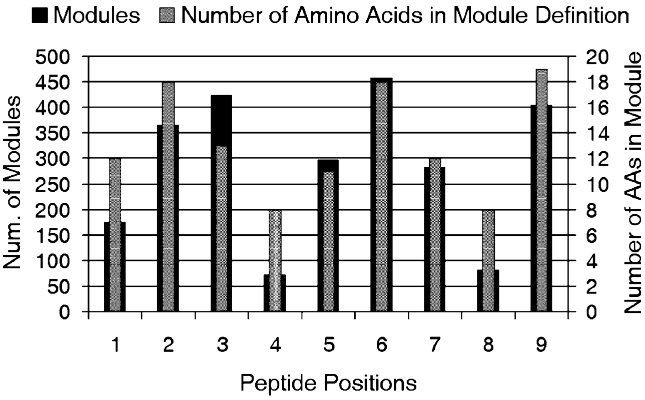


Fig. 1 Number of modules generated for each position in the peptide (*left axis*). The number of modules is dependant on the number of amino acid positions considered in the pocket definition (*right axis*), as well as the rate of polymorphism at those positions

Predictive accuracy

It was possible to calculate the performance of the matrices for 28 alleles (Table 3). In all cases, the predictive performance of the modular matrix was either within one percentage point of the control matrix or significantly better. To put these scores in context with previously published binding predictors, A_{ROC} values were generated using the established NetMHC algorithm (Nielsen et al. 2003). A local copy of NetMHC version 2.2 was evaluated using the peptide data that was applied to our own predictors. The resulting A_{ROC} values are listed in Table 3 for those alleles where a comparison with NetMHC is possible.

To test whether the modular technique can be applied to alleles for which no peptide binding data are available, matrices were generated for an allele without using the peptides that bind such allele. It was possible to generate and test such matrices for six alleles: A*0201, A*0206, B*2705, B*3501, B*5102, B*5301 (Fig. 2). All six predictions produced A_{ROC} scores greater than 0.9. A marginal drop in accuracy was observed for five alleles. For one allele, B*5102, modular prediction outperformed the standard matrix despite the fact that no B*5102 peptides were used in training. Module-based prediction demonstrated an advantage over the supertype-based prediction for B27 (B*2705) and B7 (B*3501, B*5102, B*5301) but not A2 (A*0201, A*0206).

Gained predictive power

Using a minimum of 15 peptides, prediction was possible for 28 alleles using the control matrix and 144 alleles with the modular technique (Table 4). The modular concept

Table 2 Modular matrix for A*0201

Amino acids		Positions in peptide								
		P1	P2	P3	P4	P5	P6	P7	P8	P9
A		15	3	10	5	10	10	15	10	6
C		1	0	1	2	0	2	1	1	1
D		0	0	4	5	3	2	1	1	0
E		1	0	2	12	3	2	3	6	0
F		7	0	5	1	6	4	9	5	0
G		7	0	7	12	10	4	2	8	0
H		2	0	1	1	1	1	3	3	0
I		6	10	4	3	5	7	6	3	11
K		12	0	2	8	2	2	0	5	0
L		8	62	12	5	9	12	12	10	30
M		2	8	3	0	1	2	1	1	1
N		1	0	6	2	3	2	3	3	0
P		1	0	3	11	8	10	5	5	0
Q		1	1	3	5	5	3	2	4	0
R		4	0	1	4	3	1	2	4	0
S		8	0	6	6	3	5	4	7	0
T		2	5	2	3	4	4	5	8	3
V		5	5	5	4	9	14	9	3	41
W		1	0	4	0	3	0	1	2	0
Y		7	0	6	0	3	1	2	2	0
Contributors	Number[a]	P1	P2	P3	P4	P5	P6	P7	P8	P9
A*0201	735	+	+	+	+	+	+	+	+	+
A*0202	75	+	+	−	−	−	−	−	+	−
A*0203	65	+	+	−	−	−	−	−	+	+
A*0204	38	+	+	−	+	−	−	−	−	−
A*0205	23	+	−	−	−	−	−	−	+	−
A*0206	81	+	−	−	+	+	−	+	+	+
A*0209	5	+	+	+	+	+	+	+	+	+
A*0211	4	+	+	+	+	−	−	−	−	−
A*0214	8	+	−	−	+	+	−	+	+	−
A*0207	19	−	−	−	+	+	−	+	+	+
A*0210	3	−	−	−	+	+	−	+	+	+
A*0217	1	−	−	−	+	−	−	−	−	−
A*6901	3	−	−	−	−	−	−	+	+	−
A*2603	2	−	−	−	−	−	−	−	+	−
A*6601	10	−	−	−	−	−	−	−	+	−
A*6802	40	−	−	−	−	−	−	−	+	−

Each column represents a position in the peptide. The rows are given with the one letter code for the amino acids. The values represent the frequencies of those amino acids at those positions based upon the peptides that are available for each module. The lower portion of the table shows which alleles contributed to the scores above and the [a] number of peptides used. The plus symbols indicate that this allele shares a module with A*0201 at the given peptide position.

Table 3 A_{ROC} values

	Standard matrix	Modular matrix	NetMHC
A*0201	0.94	0.94	0.96
A*0202	0.94	0.95	
A*0203	0.95	0.96	
A*0204	0.82	0.85	
A*0205	0.92	0.93	
A*0206	0.95	0.95	
A*0207	0.97	0.98	
A*0301	0.94	0.94	0.97
A*1101	0.95	0.95	0.97
A*2402	0.96	0.96	
A*2902	0.95	0.96	
A*3101	0.96	0.96	0.91
A*3301	0.93	0.93	
A*6801	0.95	0.96	
A*6802	0.93	0.94	
B*0702	0.96	0.96	0.98
B*2703	0.97	0.98	
B*2704	0.90	0.89	
B*2705	0.98	0.98	0.99
B*2706	0.94	0.96	
B*3501	0.97	0.97	
B*4002	0.95	0.98	
B*5101	0.94	0.93	
B*5102	0.92	0.94	
B*5103	0.90	0.93	
B*5301	0.97	0.96	
B*5401	0.96	0.97	
Cw*0401	0.98	0.97	

increased the number of predictable alleles from 15 (4.5%) to 75 (22.3%) of HLA-A and 12 (2.0%) to 36 (5.9%) of HLA-B proteins. The known peptides from Cw*0401 could be applied to Cw*0405, 07, and 12 as well.

Most informative alleles

Table 5 shows a ranking of alleles based upon how much new information they would provide to the modular concept if their peptides were to be made known. For the sake of clarity, only one allele from each two-digit group is listed in the table. The full rankings have been submitted as supplementary information (Supplementary Tables 1, 2, and 3). Across the three forms of ranking, 8 HLA-A alleles, 17 HLA-B alleles, and 20 HLA-C alleles are listed. A*7401 scored well in all three types of ranking, making it a particularly valuable allele to modular prediction. Of similar interest are the alleles B*4808 and Cw*1601, which ranked well in both the *maximum anchor modules* and *maximum predictables* categories.

B*4201 is an interesting candidate for peptide determination not only because its anchor modules occur 18 times among HLA proteins, making prediction possible for 11 proteins (Table 5), but also because its modules belong to multiple serological groups (Table 6). The module for P1, for example, is shared among 14 HLA*B groups: B*07, B*08, B*15, B*35, B*38, B*39, B*42, B*51, B*54, B*55, B*56, B*59, and B*67. Even the highly variable, P2 and P9 anchor modules span four and three groups, respectively. Despite this homology, insufficient peptide binding data are available for prediction. Conversely, determining the peptide binding motif of B*4201 would benefit modular binding prediction across many groups.

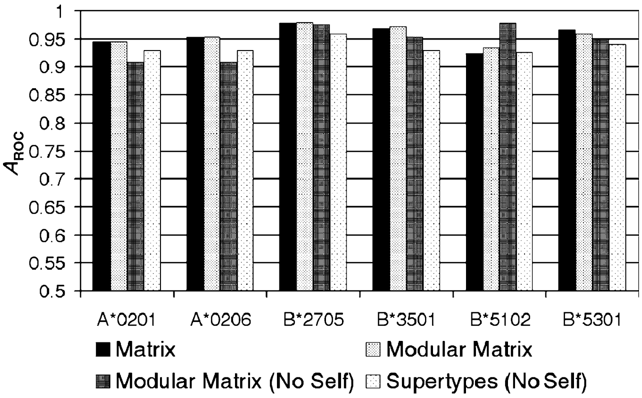


Fig. 2 Predictive performance using binding data from related alleles. The scores for the standard matrix and modular matrix predictions were generated as in Table 3. For modular matrices (no self) and supertypes (no self), scores were generated using peptide binding data from related alleles, and excluding peptide binding data for that allele

Discussion

Modules

Despite the pronounced level of diversity among HLA alleles, the alleles display a significant amount of homology on the modular level. Intergene homologies are also present, as the P7 and P8 modules for B*5302 in Table 1 demonstrate. As expected, the positions that are most important for binding (i.e. 2 and 9) and are highly variable limit the interallelic application of peptide binding data. Nonetheless, it is profitable to determine which anchor positions lack peptide binding data instead of working solely at the allele level.

The most interesting feature of Fig. 1 is the strong correlation between the number of positions involved in peptide binding and the importance of that position for the specificity of the binding. The anchor positions 2, 9, and auxiliary anchor 6 are well reflected. Each position in the binding groove carries with it a certain level of variability (an average of 3.9 amino acids per position, when considering only positions in the pocket definitions). Therefore, the number of modules found for a given pocket grows with the number of positions in its pocket definition. This is the main factor affecting the number of modules as given in Fig. 2. The other dimension that affects the number of modules is the rate of polymorphism at the positions in each module. For example, a position with very high polymorphism will disproportionally increase the number of modules for the pocket that contains such position. The biological question that arises is, "Has evolutionary pressure affected the variability of the HLA binding pocket selectively at positions involved in binding specificity?" Variability analysis of amino acids sequences can be performed using Shannon's entropy (Reche and Reinherz 2003). When analyzing modules, however, entropy calculations do not reveal a correlation with anchor positions when the entropies are averaged across all positions of the module (supplementary figure).

Prediction

The modular concept relies on the assumption that the positions in the peptide bind independently—that they are not affected by which amino acids occur at neighboring positions. Although this is not entirely the case, a great deal of independence is demonstrated by the success of the many motif- and matrix-based prediction algorithms, which do not consider such neighboring relationships. It cannot be excluded that certain module constellations create unrealistic biological environments. However, in light of the positive performance results of the modular matrix, this does not appear to be a problem.

Table 3 shows that, in all cases, the predictive performance of the modular matrix was either within one percentage point of the control matrix or significantly better. For alleles, such as A*0201, the modular

Table 4 Alleles for which prediction is possible

Standard matrix	Modular matrix			
Total=28	Total=144			
A*0201	A*0201	A*0304	A*2911	B*3507
A*0202	A*0202	A*0305	A*3101	B*3524
A*0203	A*0203	A*0306	A*3301	B*3532
A*0204	A*0204	A*0313	A*3303	B*3542
A*0205	A*0205	A*0314	A*3304	B*4002
A*0206	A*0206	A*1101	A*3305	B*4035
A*0207	A*0207	A*1102	A*3306	B*4056
A*0301	A*0209	A*1105	A*3307	B*4057
A*1101	A*0214	A*1107	A*6801	B*5101
A*2402	A*0218	A*1109	A*6802	B*5102
A*2902	A*0221	A*1112	A*6816	B*5103
A*3101	A*0222	A*1113	A*6819	B*5117
A*3301	A*0224	A*1115	A*6821	B*5118
A*6801	A*0225	A*2402	A*6822	B*5124
A*6802	A*0228	A*2405	A*6824	B*5126
B*0702	A*0230	A*2420	A*6825	B*5128
B*2703	A*0231	A*2421	A*6827	B*5130
B*2704	A*0240	A*2426	B*0702	B*5132
B*2705	A*0251	A*2427	B*0721	B*5133
B*2706	A*0258	A*2435	B*0722	B*5135
B*3501	A*0259	A*2437	B*0730	B*5301
B*4002	A*0261	A*2438	B*0733	B*5302
B*5101	A*0263	A*2439	B*0735	B*5401
B*5102	A*0266	A*2443	B*2703	B*5507
B*5103	A*0267	A*2901	B*2704	Cw*0401
B*5301	A*0268	A*2902	B*2705	Cw*0405
B*5401	A*0271	A*2906	B*2706	Cw*0407
Cw*0401	A*0272	A*2909	B*2713	Cw*0412
	A*0301	A*2910	B*3501	

Table 5 Ranking of the alleles which would provide the best new module data

Rank	Maximum modules		Maximum anchor modules		Maximum predictables	
	Allele	New module occurrences	Allele	New anchor occurrences	Allele	Newly predictable alleles
1	Cw*1502	135	Cw*1511	25	B*4808	16
2	A*3201	84	A*7401	25	B*1568	15
3	Cw*0202	84	A*3201	25	B*4028	12
4	Cw*0707	83	Cw*1601	25	B*4201	11
5	Cw*0501	79	Cw*1202	23	B*0734	11
6	A*7401	77	B*4808	22	B*4104	11
7	Cw*1208	64	Cw*0202	21	B*3533	10
8	Cw*0410	64	A*0308	20	Cw*0502	10
9	B*4101	58	Cw*0314	19	Cw*1601	10
10	Cw*0810	55	B*4201	18	Cw*0707	9
11	A*2304	55	B*0734	18	Cw*0810	9
12	B*4801	51	B*4104	18	Cw*0410	9
13	Cw*1701	48	B*5605	17	Cw*1203	8
14	A*3108	47	Cw*1701	16	A*7401	8
15	B*5518	47	B*4028	16	B*4901	7

Based on the modular technique, a ranking of the alleles can be made, which reflects the amount of new information that they would provide if their binders were purified and sequenced. This list provides a way of prioritizing which alleles should be studied next. Alleles at the top of the *maximum module occurrences* list contain the highest number of modules that are shared by the most other unstudied alleles. Similarly, the *maximum anchor occurrences* list is based upon the highest number of unstudied anchor positions (P2 and P9). The *maximum predictable alleles* list ranks the alleles which would maximize the number of alleles for which binding data for P2 and P9 are simultaneously available (thereby making the alleles predictable). For the sake of clarity, only one allele from each two-digit group is listed.

matrix hardly differs from the standard matrix. This is because the majority of the peptide binding data comes directly from A*0201 binders—The few peptides from related alleles (A*0202, A*0203, etc) are not numerous enough to significantly influence the matrix. Although this is the case for several of the alleles in Table 3, it is interesting to note that contributions from related alleles were sometimes helpful but never harmful to prediction accuracy.

To orient our results on an established HLA binding predictor, we tested NetMHC with our peptide data. The A_{ROC} scores for NetMHC with the MHCBN peptides were very good for A*0201, A*0301, A*1101, A*3101, B*0702, and B*2705 (see Predictive accuracy), demonstrating the effectiveness of this technique when one considers that NetMHC was not trained on exactly this set of peptide binding data. The NetMHC algorithm outperformed the matrix and modular matrix in every case except B*3101. NetMHC's poorer performance for B*3101 can be explained by the presence of a secondary P9 anchor of lysine in the MHCBN, which is absent in the training data for NetMHC. The fact that the modular matrix is not able to outperform the neural network-based approach in most cases is not surprising, and it should be emphasized that the purpose of this experiment is to expand the number of predictable HLA alleles using the structural data provided in the pocket definitions.

The results of the "no self" analysis (Fig. 2) demonstrate that peptide binding prediction for an allele is possible using binding data only from other alleles. Modular prediction for B*5102 significantly outperformed the standard matrix despite the fact that no B*5102 peptides were used in training. This can be attributed to the fact that there are 32 binding nonamers available for B*5102, but by utilizing modular data, hundreds of peptides are considered: 343 peptides at P1, 445 at P2, 224 at P3, 708 at P4, and 224 at P5–P9.

The comparison of the modular matrix-based "no self" analysis with the supertypes "no self" analysis show that our technique offers an improvement over B7 and B27 supertypes but not the A2 supertype. The case of A2 exemplifies that our approach to generating modules is at times too strict—A single amino acid mismatch, even between functionally similar residues, results in two distinct modules, and as such, the one module cannot benefit from the peptide binding data of the other. However, it is likely this strictness that leads to the improvement in prediction over the B7 and B27 supertypes. For example, the module matrix for B*5102 includes binding P1 and P2 binding data from B*3501 but not for other positions. It is particularly important that the P9 motif of B*3501 (L/M/F/Y) is excluded from the B*5102 prediction, because it differs significantly from B*5102's P9 motif (I/L/V). The absence of such

Table 6 Modules for B*4201

		Known peptides
P1 pocket	1, 5, 7, 33, 59, 62, 63, 66, 99, 159, 163, 167, 171	180
Module	MYFYRNIYYRWY	
Alleles	B*0719, 31, 34, 43, B*0801, 02, 04, 06, 07, 09, 12–16, 18, 20, 22–24, B*1544, 93, B*1811, B*3535, 60, B*3801, 02, 05–07, 09–11, B*3901, 03–06, 10, 12, 14–20, 24, 26–31, 34, B*4201, 02, 04–06, B*5136, B*5401, 02, 04, 07, B*5501–05, 07, 10–17, 19, B*5610, 12, B*5901, B*6701	
P2 pocket	2, 7, 9, 24, 25, 26, 34, 35, 36, 45, 62, 63, 66, 67, 70, 99, 159, 163, 167	3
Module	YYSVGVRFERNIYQYYTW	
Alleles	B*0719, 31, 34, 43, B*4201, 04–06, B*5510, B*6701	
P3 pocket	3, 7, 9, 62, 66, 70, 97, 99, 114, 152, 155, 156, 159, 163	1
Module	YYRIQSYNVQDYT	
Alleles	B*4201, 05, 06	
P4 pocket	4, 62, 65, 66, 69, 70, 155, 156, 159	1
Module	RQIAQQDY	
Alleles	B*0704, 19, 25, B*4201, 02, 04–06, B*4506, B*5613, B*8201, 02, B*8301	
P5 pocket	5, 69, 70, 73, 74, 97, 114, 116, 152, 155, 156, 159	1
Module	AQTDSNYVQDY	
Alleles	B*4201, 02, 05	
P6 pocket	6, 7, 9, 22, 24, 66, 69, 70, 73, 74, 97, 99, 114, 116, 133, 147, 152, 155, 156	
Module	YYFSIAQTDSYNYWWVQD	
Alleles	B*4201, 05	
P7 Pocket	7, 73, 77, 97, 114, 116, 133, 146, 147, 150, 152, 155, 156	76
Module	TSSNYWKWAVQD	
Alleles	B*0801, 04, 05, 10, 11, 15, 18, 21–24 B*4102 B*4201, 02, 05	
P8 pocket	8, 73, 76, 77, 80, 97, 143, 146, 147	318
Module	TESNSTKW	
Alleles	B*0702–06, 08–10, 16, 17, 19–26, 28–35, 37, 39–43, B*0801, 04, 05, 07, 10, 11, 13, 14, 18, 20–25, B*1405, B*1507, 45, 55, 68, B*1814, B*3505, 16, 17, 22, 30, 31, 51, 58, B*3903, 14, 24, 29, B*4002, 03, 05, 08, 09, 15, 16, 18, 24, 27, 29, 32, 35, 39, 40, 50, 56–58, B*4102, 04, B*4201, 02, 05, 06, B*48, 08, 10, 12, 13, B*5504	
P9 pocket	9, 70, 73, 74, 76, 77, 80, 81, 84, 95, 96, 97, 114, 116, 123, 124, 142, 143, 146, 147	1
Module	QTDESNLYLQSNYYIITKW	
Alleles	B*0705, 06, 34, 40, B*4201, 02, 05, B*5504	

HLA-B*4201 is an example of a highly informative allele, which would contribute significant modular peptide binding data if it were to be analyzed. The nine modules here are lists of amino acids from B*4201 that play a role in binding a particular position in a nonamer peptide. The rows "P[1–9] pocket" contain the definitions of which amino acids positions are responsible for binding the respective position in the peptide according to Chelvanayagam. Each module form B*4201 occurs in other alleles as well. These alleles are listed in the rows designated "Alleles." The numbers of peptides that are associated with each module are listed in the "Known peptides" column.

selective exclusion is likely responsible for the slightly lower performance of the supertype-based prediction in this case. The modular matrix's P9 motif is that of B*5101 (I/L/V), which is identical to B*5102.

In conclusion, the modular approach provides a high specificity when deciding how to employ peptide data of related alleles, and the supertype approach a high sensitivity. The modular approach addresses the problem of classifying alleles that, on one side of the binding groove, fit into one supertype and on the other side of the binding groove, in a different supertype. The modular approach does not make use of known binding motifs, which is a valuable source of information for the generation of supertypes (Sette and Sidney 1999). This is, at the same time, an advantage because such binding motifs are not always available.

Most informative alleles

Table 5 shows a ranking of alleles based upon how much new information they would provide to the modular concept if their peptides were to be made known. Although maximizing the number of modules is productive for completeness, it is not perfect for maximizing predictive capabilities. This is because of the proportional relationship between the variability of an amino acid position in HLA and that position's significance for peptide binding (Reche and Reinherz 2003). The modules that are shared among the most alleles are found in the least polymorphic areas and, therefore, have a minimal effect on peptide binding. For example, P8 plays a minor role in peptide binding, and its modules are shared amongst the most alleles. Maximizing the anchor positions ensures that the newly provided information is relevant for peptide binding. The alleles in this list should be studied in the long-term interest of comprehensive peptide binding.

The question of which alleles to study to maximize the number of predictable alleles in one step is answered in the "Maximum predictables" column (Table 5). This list considers previous peptide binding data, which could be used in combination with new data to maximize the number of predictable alleles in the short term. For example, peptide binding data for B*4808 would make the prediction of B*4009 possible by providing data for the module at P9, which these two alleles have in common. Peptide binding data are already available for B*4009's P2 anchor position via B*4001 and B*4002. In this way, previous data could be combined efficiently with new binding data to maximize the number of predicable alleles. Additionally, studying these alleles is also useful for further confirmation or refutation of the modular concept of HLA.

Application in immunotherapy

Because of HLA diversity, individualized immunotherapy may offer leukemia patients the best chances for preventing relapse without developing GvHD in hematopoietic stem cell transplantation protocols. This involves stimulating donor T cells to react with the minor histocompatibility antigens, which are presented on the surface of the patient's malignant cells but are not presented in GvHD-susceptible tissues. HLA peptide binding prediction plays a central role in identifying which peptides can be used as T cell targets to produce a GvL effect. To accurately predict T cell targets, a system involving TAP binding prediction and proteasomal processing prediction is necessary as well (Bhasin and Raghava 2004; Donnes and Kohlbacher 2005; Doytchinova and Flower 2006; Guan et al. 2006; Larsen et al. 2005; Tenzer et al. 2005; Zhang et al. 2006). Incorporated into such a system, the modular concept of HLA is a promising step in making peptide binding prediction for all patients a reality. Further information on the modular model of HLA, as well as tools for finding personalized alloreactive peptides, can be found on the Internet at http://www.peptidecheck.org.

Acknowledgement The MHCBN database was kindly and conveniently provided by GPS Raghava and Manoj Bhasin of the Institute of Microbial Technology, Chandigarh, India. The NetMHC algorithm and technical support were provided by Søren Brunak and Claus Lundegaard of the Center for Biological Sequence Analysis, BioCentrum-DTU, Denmark. The authors are very grateful to Peter Horn of the Institute for Transfusion Medicine at Hanover Medical School for bringing important perspectives to the analysis of these results.

References

Bade-Doeding C, Elsner HA, Eiz-Vesper B, Seltsam A, Holtkamp U, Blasczyk R (2004) A single amino-acid polymorphism in pocket A of HLA-A*6602 alters the auxiliary anchors compared with HLA-A*6601 ligands. Immunogenetics 56:83–88

Bade-Doeding C, Eiz-Vesper B, Figueiredo C, Seltsam A, Elsner HA, Blasczyk R (2005) Peptide-binding motif of HLA-A*6603. Immunogenetics 56:769–72

Bhasin M, Raghava GP (2004) Analysis and prediction of affinity of TAP binding peptides using cascade SVM. Protein Sci 13:596–607

Bhasin M, Singh H, Raghava GP (2003) MHCBN: a comprehensive database of MHC binding and non-binding peptides. Bioinformatics 19:665–666

Buus S, Lauemoller SL, Worning P, Kesmir C, Frimurer T, Corbet S, Fomsgaard A, Hilden J, Holm A, Brunak S (2003) Sensitive quantitative predictions of peptide-MHC binding by a 'Query by Committee' artificial neural network approach. Tissue Antigens 62:378–384

Chelvanayagam G (1996) A roadmap for HLA-A HLA-B and HLA-C peptide binding specificities. Immunogenetics 45:15–26

Davies MN, Sansom CE, Beazley C, Moss DS (2003) A novel predictive technique for the MHC class II peptide-binding interaction. Mol Med 9:220–225

Donnes P, Elofsson A (2002) Prediction of MHC class I binding peptides using SVMHC. BMC Bioinformatics 3:25

Donnes P, Kohlbacher O (2005) Integrated modeling of the major events in the MHC class I antigen processing pathway. Protein Sci 14:2132–2140

Doytchinova IA, Flower DR (2006) Class I T-cell epitope prediction: improvements using a combination of proteasome cleavage TAP affinity and MHC binding. Mol Immunol 43:2037–2044

Gotch F, McMichael A, Rothbard J (1988) Recognition of influenza A matrix protein by HLA-A2-restricted cytotoxic T lymphocytes. Use of analogues to orientate the matrix peptide in the HLA-A2 binding site. J Exp Med 168:2045–2057

Goulmy E, Schipper R, Pool J, Blokland E, Falkenburg JH, Vossen J, Gratwohl A, Vogelsang GB, van Houwelingen HC, van Rood JJ (1996) Mismatches of minor histocompatibility antigens between HLA-identical donors and recipients and the development of graft-versus-host disease after bone marrow transplantation. N Engl J Med 334:281–285

Guan P, Hattotuwagama CK, Doytchinova IA, Flower DR (2006) MHCPred 2.0: an updated quantitative T-cell epitope prediction server. Appl Bioinformatics 5:55–61

Hambach L, Goulmy E (2005) Immunotherapy of cancer through targeting of minor histocompatibility antigens. Curr Opin Immunol 17:202–210

Kotsch K, Blasczyk R (2000) The noncoding regions of HLA-DRB uncover interlineage recombinations as a mechanism of HLA diversification. J Immunol 165:5664–5670

Larsen MV, Lundegaard C, Lamberth K, Buus S, Brunak S, Lund O, Nielsen M (2005) An integrative approach to CTL epitope prediction: a combined algorithm integrating MHC class I binding TAP transport efficiency and proteasomal cleavage predictions. Eur J Immunol 35:2295–2303

Nielsen M, Lundegaard C, Worning P, Lauemoller SL, Lamberth K, Buus S, Brunak S, Lund O (2003) Reliable prediction of T-cell epitopes using neural networks with novel sequence representations. Protein Sci 12:1007–1017

Noguchi H, Kato R, Hanai T, Matsubara Y, Honda H, Brusic V, Kobayashi T (2002) Hidden Markov model-based prediction of antigenic peptides that interact with MHC class II molecules. J Biosci Bioeng 94:264–270

Parker KC, Bednarek MA, Coligan JE (1994) Scheme for ranking potential HLA-A2 binding peptides based on independent binding of individual peptide side-chains. J Immunol 152:163–75

Rammensee H, Bachmann J, Emmerich NP, Bachor OA, Stevanovic S (1999) SYFPEITHI: database for MHC ligands and peptide motifs. Immunogenetics 50:213–219

Reche PA, Reinherz EL (2003) Sequence variability analysis of human class I and class II MHC molecules: functional and structural correlates of amino acid polymorphisms. J Mol Biol 331:623–641

Reche PA, Glutting JP, Zhang H, Reinherz EL (2004) Enhancement to the RANKPEP resource for the prediction of peptide binding to MHC molecules using profiles. Immunogenetics 56:405–419

Robinson J, Waller MJ, Parham P, de Groot N, Bontrop R, Kennedy LJ, Stoehr P, Marsh SG (2003) IMGT/HLA and IMGT/MHC: sequence databases for the study of the major histocompatibility complex. Nucleic Acids Res 31:311–314

Rognan D, Scapozza L, Folkers G, Daser A (1994) Molecular dynamics simulation of MHC–peptide complexes as a tool for predicting potential T cell epitopes. Biochemistry 33:11476–11486

Rothbard JB (1992) Synthetic peptides as vaccines. Biotechnology 20:451–465

Sette A, Sidney J (1999) Nine major HLA class I supertypes account for the vast preponderance of HLA-A and -B polymorphism. Immunogenetics 50:201–212

Sturniolo T, Bono E, Ding J, Raddrizzani L, Tuereci O, Sahin U, Braxenthaler M, Gallazzi F, Protti MP, Sinigaglia F, Hammer J (1999) Generation of tissue-specific and promiscuous HLA ligand databases using DNA microarrays and virtual HLA class II matrices. Nat Biotechnol 17:555–561

Tenzer S, Peters B, Bulik S, Schoor O, Lemmel C, Schatz MM, Kloetzel PM, Rammensee HG, Schild H, Holzhutter HG (2005) Modeling the MHC class I pathway by combining predictions of proteasomal cleavage TAP transport and MHC class I binding. Cell Mol Life Sci 62:1025–1037

Yanover C, Hertz T (2005) Predicting protein–peptide binding affinity by learning peptide–peptide distance functions. Lect Notes Comput Sci 3500:456–471

Yewdell JW, Bennink JR (1999) Immunodominance in major histocompatibility complex class I-restricted T lymphocyte responses. Annu Rev Immunol 17:51–88

Zhang GL, Petrovsky N, Kwoh CK, August JT, Brusic V (2006) PREDTAP: a system for prediction of peptide binding to the human transporter associated with antigen processing. Immunome Res 2:3

Zhu S, Udaka K, Sidney J, Sette A, Aoki-Kinoshita KF, Mamitsuka H (2006) Improving MHC binding peptide prediction by incorporating binding data of auxiliary MHC molecules. Bioinformatics 22:1648–55

8 Implementing the Modular MHC Model

Title

Implementing the Modular MHC Model for Predicting Peptide Binding

Authors

DeLuca, D.S. and Blasczyk, R.

Published by invitation in

Immunoinformatics (Methods in Molecular Biology), 2007

18

Implementing the Modular MHC Model for Predicting Peptide Binding

David S. DeLuca and Rainer Blasczyk

Summary

The challenge of predicting which peptide sequences bind to which major histocompatibility complex (MHC) molecules has been met with various computational techniques. Scoring matrices, hidden Markov models, and artificial neural networks are examples of algorithms that have been successful in MHC–peptide-binding prediction. Because these algorithms are based on a limited amount of experimental peptide-binding data, prediction is only possible for a small fraction of the thousands of known MHC proteins. In the primary field of application for such algorithms—vaccine design—the ability to make predictions for the most frequent MHC alleles may be sufficient. However, emerging applications of leukemia-specific T cells require a patient-specific MHC–peptide-binding prediction. The modular model of MHC presented here is an attempt to maximize the number of predictable MHC alleles, based on a limited pool of experimentally determined peptide-binding data.

Key Words: Modules; pockets; HLA; MHC; class I; class II; peptide; binding; prediction

1. Introduction

The major histocompatibility complex (MHC) is a highly polymorphic collection of genes encoding membrane surface proteins, which plays an important role in the immune system. MHC binds short peptide sequences and presents them on the cell surface for inspection by T cells *(1)*. In humans, MHC is known as human leukocyte antigen (HLA).

Because of MHC's role in recognizing pathogenic and cancerous peptides, these genes are under high environmental pressure to be very polymorphic.

From: *Methods in Molecular Biology, vol. 409: Immunoinformatics: Predicting Immunogenicity In Silico*
Edited by: D. R. Flower © Humana Press Inc., Totowa, NJ

Presently, 2,088 HLA alleles have been identified *(2)*. Predicting which peptide sequences will bind to specific MHC alleles is dependent on the amount of experimentally determined peptide-binding data available for each allele. Such data are only available for a small fraction of all the alleles. The goal of the modular concept is to take advantage of similarities among alleles by utilizing existing peptide-binding data to make predictions for alleles, for which no peptides are available.

Although MHC polymorphism can be caused by point mutation, it is mainly a result of gene conversion and recombination *(3)*. Therefore, although a specific MHC is unique, it may be identical to a second MHC in one region and identical to a third MHC in another region. Such similarities can be exploited by breaking down MHC into modules and correlating these modules with the available peptide-binding data *(4,5)*. In this way, peptide-binding data specific for a small number of MHC variants can be applied to an expanded number of variants.

The part of the MHC–peptide-binding groove that interacts with a specific position in the bound peptide is known as a pocket. Originally these pockets were designated A–F *(6)*. Further analysis of crystallographic data in class I HLA has provided more complete definitions of which positions in HLA are responsible for binding certain positions in the peptide *(7,8)*. Because of the side chain orientation in the protein's three-dimensional structure, the positions responsible for peptide binding are not sequential. For example, the particular residues in HLA class I that interact with the N-terminal amino acid (P1 = peptide position 1) in the peptide are at positions 5, 7, 33, 59, 62, 63, 66, 99, 159, 163, 167, and 171 *(7)*. These positions are used to define a module. A module is the sequence of amino acids found at these positions in a specific MHC allele. For a 9-mer peptide, a given allele will have nine modules (P1, P2,... P9). Because of similarities among MHC alleles, different MHCs can share modules when they posses the same amino acids at the defined positions (Tables 1 and 2).

The result of this modular concept is an expanded number of MHC alleles, for which peptide binding can be predicted.

2. Implementation

The modular prediction algorithm available via the PeptideCheck (http://www.peptidecheck.org) website was written in Java and runs on a Tomcat application server, utilizing servlets, java server pages, and a MySQL database.

Table 1
Modules for A*0101 and A*7401 at P1

A*0101												
Position	5	7	33	59	62	63	66	99	159	163	167	171
Amino acid	M	Y	F	Y	Q	E	N	Y	Y	R	G	Y
Other alleles with this module:					A*0102, A*0103, A*0106, A*0107, A*0110							
A*7401												
Position	5	7	33	59	62	63	66	99	159	163	167	171
Amino acid	M	Y	F	Y	Q	E	N	Y	Y	T	W	Y
Other alleles with this module:	A*0256, A*0301-14, A*1104, A*3001–6, 8, 9, 11, 12, A*3101, 3, 4, 6, 9, A*3201–4, 6–8, A*3601–3, A*7402, 3, 5–10											

The positions listed here are positions in the HLA protein, which are likely to affect the binding of amino acids at P1 in the peptide. The amino acids listed are those amino acids which occur at the given positions in A*0101 and A*7401, respectively. These lists of nonsequential amino acids are the modules at P1. The alleles listed under "Other alleles with this module" possess the same amino acids at these positions and therefore possess the same P1 modules.

Table 2
Number of modules for each peptide position

Peptide positions	1	2	3	4	5	6	7	8	9
Number of modules	176	365	424	72	298	458	282	82	405

The total number of modules for each peptide position is less than the number of HLA proteins, because related alleles share certain modules. These numbers are based on all class I HLA-A, HLA-B, and HLA-C proteins from the IMGT/HLA database version 2.10.0, which contains 1,098 class I proteins.

2.1. HLA Sequence Data

HLA protein sequences are available in the IMGT/HLA database and are regularly updated *(2)*. Sequences can be downloaded directly from the file transfer protocol (FTP) server under *ftp://ftp.ebi.ac.uk/pub/databases/imgt/mhc/hla/*. Nucleotide and protein sequences are available in various formats. Sequence alignments for all HLA genes are available as zip files. Because many of the HLA sequences are incomplete (e.g., only certain exons have been determined), sequence alignments are necessary. Programmers may either download the individual sequences, and align them locally, or download the alignment files, and extract the sequence information.

2.2. Peptides

The module-based peptide-binding prediction requires collections of peptide, which have been experimentally proven to bind MHC. Databases such as SYFPEITHY, MHCBN, and AntiJen are good sources of peptide-binding data ***(9–11)***. Although some databases provide binding affinities, the algorithms described here require only that a distinction is made between binders and nonbinders. Nonbinders are often a limiting factor. Alternatively, random sequences of peptides can be generated and assumed to be nonbinders. This assumption will be true for the vast majority of sequences because less than 1% of possible peptide sequences are thought to bind HLA class I ***(12)***. The use of random nonbinders has several precedents ***(13,14)***. In this implementation, random nonamers were generated by randomly choosing human proteins from the Entrez protein database. Segments of nine amino acids were then randomly chosen.

2.3. Modules

At the heart of the modular concept lies the pocket definition. For our purposes, a pocket is the list of positions in HLA, which is responsible for binding a particular amino acid position in the peptide. In this study, the pockets were defined as per Chelvanayagam's analysis of crystallographic HLA data ***(7)***. Alternative definitions have been provided by Saper and Reche ***(6,8)***.

A module is the sequence of amino acids found at the pocket positions for a given allele. Modules are generated by combining the pocket definitions provided by Chelvanayagam or others with the HLA protein sequences (Table 1). Although many related alleles produce the same module sequences, only unique sequences should be stored in the database. A second database table can be used to correlate the module sequences with the alleles that posses them.

2.4. Matrices and Prediction

The simplest implementation of modular peptide-binding prediction is using a scoring matrix. When predicting binding to nonamers, the matrices are 9 × 20 and contain values for each amino acid at each position peptide (Table 3). The following pseudocode demonstrates how to generate the matrix:

Modular Matrix

```
For each module
    Retrieve all alleles that have this module
    For all alleles with this module
        Retrieve all binders
        For each binder
            Count the amino acid at the position corresponding to
            this module
    Divide the scores by the number of binders found for this module
```

A score for a peptide's binding ability is generated by multiplying the nine corresponding values from the matrix. This score is indicative of the likelihood that this peptide is a binder and can be compared to a threshold to predict binding. In the modular matrix, the values are based on the frequencies of the amino acids, among binding peptides, specific to a particular module (*see* **Note 1**). Because different alleles can have certain modules in common, the module-specific values are based on peptides that bind to all the alleles which have that module.

2.5. Evaluating Predictive Performance

Predictive performance can be calculated using the area under the receiver operating characteristic curve (A_{ROC}). The ROC curve is based on the prediction's sensitivity

$$\mathrm{SE} = {}^{\mathrm{TP}}/_{(\mathrm{TP}+\mathrm{FN})}$$

and specificity

$$\mathrm{SP} = {}^{\mathrm{TN}}/_{(\mathrm{TN}+\mathrm{FP})}$$

where TP = true positives—correctly predicted binders; FN = false negatives—binders incorrectly predicted to be nonbinders; TN = true negatives—correctly predicted nonbinders; and FP = false positives—nonbinders incorrectly predicted to bind. The ROC curve is a plot of SE versus 1 SP over a range of thresholds (Fig. 1).

Using the same peptides for training as well as testing is for obvious reasons taboo. Peptides used in testing should be excluded from the matrix scores. This can be done by splitting the peptide data into separate training and testing pools (e.g., two-thirds for training and one-third for testing). A method that

Table 3
Modular matrix for A*0201

Amino acids	Positions in peptide								
	P1	P2	P3	P4	P5	P6	P7	P8	P9
A	15	3	10	5	10	10	15	10	6
C	1	0	1	2	0	2	1	1	1
D	0	0	4	5	3	2	1	1	0
E	1	0	2	12	3	2	3	6	0
F	7	0	5	1	6	4	9	5	0
G	7	0	7	12	10	4	2	8	0
H	2	0	1	1	1	1	3	3	0
I	6	10	4	3	5	7	6	3	11
K	12	0	2	8	2	2	0	5	0
L	8	62	12	5	9	12	12	10	30
M	2	8	3	0	1	2	1	1	1
N	1	0	6	2	3	2	3	3	0
P	1	0	3	11	8	10	5	5	0
Q	1	1	3	5	5	3	2	4	0
R	4	0	1	4	3	1	2	4	0
S	8	0	6	6	3	5	4	7	0
T	2	5	2	3	4	4	5	8	3
V	5	5	5	4	9	14	9	3	41
W	1	0	4	0	3	0	1	2	0
Y	7	0	6	0	3	1	2	2	0

Contributors	Num.	P1	P2	P3	P4	P5	P6	P7	P8	P9
A*0201	735	+	+	+	+	+	+	+	+	+
A*0202	75	+	+	−	−	−	−	−	+	−
A*0203	65	+	+	−	−	−	−	−	+	+
A*0204	38	+	+	−	+	−	−	−	−	−
A*0205	23	+	−	−	−	−	−	−	+	−
A*0206	81	+	−	−	+	+	−	+	+	+
A*0209	5	+	+	+	+	+	+	+	+	+
A*0211	4	+	+	+	+	−	−	−	−	−
A*0214	8	+	−	−	+	+	−	+	+	−
A*0207	19	−	−	−	+	+	−	+	+	+
A*0210	3	−	−	−	+	+	−	+	+	+
A*0217	1	−	−	−	+	−	−	−	−	−
A*6901	3	−	−	−	−	−	−	+	+	−
A*2603	2	−	−	−	−	−	−	−	+	−
A*6601	10	−	−	−	−	−	−	−	+	−
A*6802	40	−	−	−	−	−	−	−	+	−

Each column represents a position in the peptide. The rows are given with the one-letter code for the amino acids. The values represent the frequencies of those amino acids at those positions, based on the peptides that are available for each module. The lower portion of the table shows which alleles contributed to the scores above and the number of peptides (Num.) used. The "+" symbol indicates that this allele shares a module with A*0201 at the given peptide position.

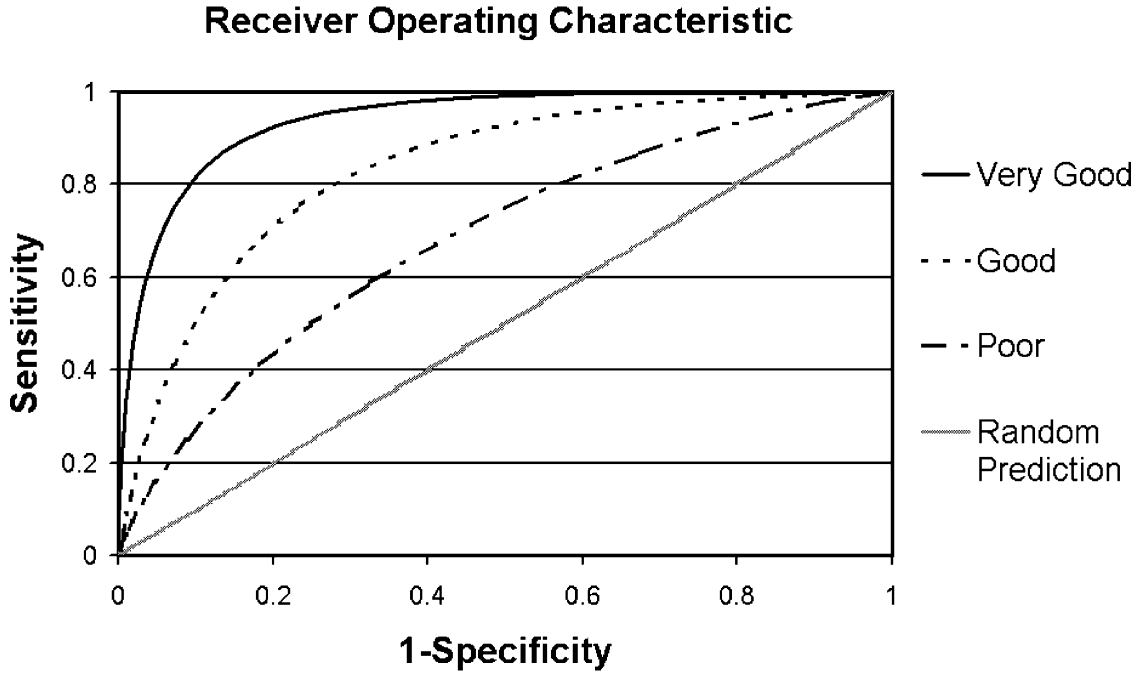


Fig. 1. Receiver operating characteristic (ROC) curves. The ROC curve is a function of specificity as well as sensitivity. The area under the ROC curve (A_{ROC}) is the standard measure of accuracy for major histocompatibility complex (MHC)–peptide-binding prediction. Random prediction refers to the expected results when randomly guessing whether the peptide is a binder or nonbinder.

delivers better result, especially when few peptides are available, but is more computationally intensive is the "jackknife" technique. Before performing the prediction for a given peptide, the peptide and all peptides with only one amino acid difference are removed from the training data, and the matrices were calculated without these peptides.

A goal of the modular concept is to make prediction possible for alleles, for which no peptide data are available. To test the modular concept, a "no-self" evaluation is necessary. In this implementation, the values in the modular matrix were generated and tested for a given allele, without using peptide-binding data for that allele. For example, predictions were made for A*0201 using binding data from other alleles (A*0202–0206, 0209, 0211, 0214, 0207, 2603, 6601, 6802, and 6901) but excluding peptides proven to bind A*0201.

3. Application

The module-based HLA–peptide-binding prediction is available as part of the PeptideCheck website (http://www.peptidecheck.org).

3.1. Predicting HLA–peptide Binding

In the simplest case, the user can enter a peptide sequence and choose an HLA allele. The result is a score representing the probability that the given peptide is bound by the given allele. Alternatively, the user may enter a protein sequence, and all possible resulting peptides are scored. Conveniently, more than one HLA allele can be chosen at a time.

The prediction algorithm generates a score. To determine whether this score is indicative of binding or nonbinding, it must be compared to a threshold. Choosing a threshold is dependent on experimental context. For example, if the user is intent on finding peptides that will have the highest chance of binding in the laboratory, a very high threshold is recommended. If the question is whether a peptide is or is not a minor histocompatibility antigen (peptide derived from a variant region of a non-HLA protein) then a balanced threshold is necessary. The threshold suggested in the PeptideCheck website is the point at which the sensitivity and specificity curves cross. Unfortunately, it is not possible to suggest thresholds for all predictable alleles. One can only generate sensitivity and specificity curves when peptide-binding data are available. However, modular peptide-binding prediction allows for prediction when no data are available (*see* **Note 2**). In this case, no threshold can be suggested, and it is recommended that the user compares scores to find peptides that represent the most likely binders.

3.2. Predicting Peptide Presentation Profile/Individual's Peptide-binding characteristics

In the area of leukemia-specific T-cell therapy, it is important to compare the peptide-binding profile of the patient. Peptide-binding profiles can be created by entering the patient's HLA genotype. In the case of a full heterozygosity, this includes two alleles from each of the HLA-A, HLA-B, and HLA-C loci. The user can either provide a peptide, one or more protein sequences, or a single-nucleotide polymorphism (SNP) profile for analysis. The resulting table displays the best binders, the proteins that they stem from, the binding score, and to which alleles they bind.

3.3. Exploring Modular Relations Between HLA Alleles

To understand the relations between various HLA alleles, it can be useful to compare them at the modular level. This is particularly useful when choosing which HLA alleles to study when determining peptide-binding motifs. After selecting an allele, the user is presented with the list of modules that this allele possesses. Clicking on a module brings up the list of alleles that possess this module. If binding motifs are available, they are also displayed. In this way, the user can choose an allele and find information about its binding motif based on the binding data for other alleles. Conversely, the user may determine which other alleles would benefit from the binding data of the target allele, if its peptides were to be purified and sequenced. In this way, researchers can choose those alleles for study, which are the most informative on a modular level. Prioritizing alleles in this way will ensure that peptide-binding data be found most efficiently to maximize modular peptide prediction.

Notes

1. Although the modular concept of HLA has been shown to be successful in expanding the number of predictable HLA alleles, the implementation described here has several drawbacks. The matrix scores are based on the assumption that there is a correlation between the rate of occurrence of particular amino acids at particular positions in the peptides and the importance of those amino acids in peptide binding. Although this may be true for pool sequences, many of the peptides in the peptide databases are of synthetic origin. The synthetic peptides are based on known binders but contain specific amino acid substitutions, with the goal of uncovering the roles of certain positions in the peptide. These synthetic peptides invalidate the assumption mentioned above. Drawing a correlation between peptide sequences and binding affinity is certainly a solution to this problem.
2. The modular concept will be expanded in the future to make prediction possible for more alleles, through the clustering of modules. There are module sequences that differ only slightly from each other, and which bind the same amino acids, despite small differences. Such modules will be clustered together in future implementations to maximize the usability of the provided peptide-binding data. Module-based supertypes are also an interesting consequence of such an analysis.

References

1. Marsh, S. G., Parham, P. & Barber, L. D. (2000). *The HLA FactsBook*. Academic Press, London.
2. Robinson, J., Waller, M. J., Parham, P., de Groot, N., Bontrop, R., Kennedy, L. J., Stoehr, P. & Marsh, S. G. (2003). IMGT/HLA and IMGT/MHC: sequence databases for the study of the major histocompatibility complex. *Nucleic Acids Res* 31, 311–4.
3. Kotsch, K. & Blasczyk, R. (2000). The noncoding regions of HLA-DRB uncover interlineage recombinations as a mechanism of HLA diversification. *J Immunol* 165, 5664–70.

4. Bade-Doeding, C., Eiz-Vesper, B., Figueiredo, C., Seltsam, A., Elsner, H. A. & Blasczyk, R. (2005). Peptide-binding motif of HLA-A*6603. *Immunogenetics* 56, 769–72.
5. DeLuca, D. S., Khattab, B. & Blasczyk, R. (2007). A modular concept of HLA for comprehensive peptide binding prediction. *Immunogenetics* 59, 25–35.
6. Saper, M. A., Bjorkman, P. J. & Wiley, D. C. (1991). Refined structure of the human histocompatibility antigen HLA-A2 at 2.6 A resolution. *J Mol Biol* 219, 277–319.
7. Chelvanayagam, G. (1996). A roadmap for HLA-A, HLA-B, and HLA-C peptide binding specificities. *Immunogenetics* 45, 15–26.
8. Reche, P. A. & Reinherz, E. L. (2003). Sequence variability analysis of human class I and class II MHC molecules: functional and structural correlates of amino acid polymorphisms. *J Mol Biol* 331, 623–41.
9. Rammensee, H., Bachmann, J., Emmerich, N. P., Bachor, O. A. & Stevanovic, S. (1999). SYFPEITHI: database for MHC ligands and peptide motifs. *Immunogenetics* 50, 213–9.
10. Bhasin, M., Singh, H. & Raghava, G. P. (2003). MHCBN: a comprehensive database of MHC binding and non-binding peptides. *Bioinformatics* 19, 665–6.
11. Blythe, I. A. D., & Flower, D. R. (2001). JenPep: a database of quantitative functional peptide data for immunology. *Bioinformatics* 18, 434–9.
12. Yewdell, J. W. & Bennink, J. R. (1999). Immunodominance in major histocompatibility complex class I-restricted T lymphocyte responses. *Annu Rev Immunol* 17, 51–88.
13. Reche, P. A., Glutting, J. P., Zhang, H. & Reinherz, E. L. (2004). Enhancement to the RANKPEP resource for the prediction of peptide binding to MHC molecules using profiles. *Immunogenetics* 56, 405–19.
14. Donnes, P. & Elofsson, A. (2002). Prediction of MHC class I binding peptides, using SVMHC. *BMC Bioinformatics* 3, 25.

9 Minor Histocompatibility Antigen Prediction

Title

High thoughput minor histocompatibility antigen prediction

Authors

DeLuca, D.S., Eiz-Vesper, B., Ladas, N., Khattab, B., and Blasczyk, R.

Prepared for

Bioinformatics, 2008

High throughput minor histocompatibility antigen prediction

David S. DeLuca, Britta Eiz-Vesper, Nektarios Ladas, Barbara Anna-Maria Khattab, Rainer Blasczyk *

[1]Institute for Transfusion Medicine, Hannover Medical School, Carl-Neuberg-Str. 1, Hanover, Germany

ABSTRACT

The search for minor histocompatibility antigens (mHags) has implications not only for preventing graft versus host disease, but also for therapeutic applications involving leukaemia-specific T cells. We have created a web-based system, named PeptideCheck, for analyzing peptide elution data to search for mHags as well as for prediction mHags from polymorphism and protein databases. Comparison with known mHag data reveals that some but not all of the previously known mHags can be reproduced. By applying a system of filtering and ranking, we were able to produce an ordered list of potential mHag candidates in which HA-1, HA-3, and HA-8 occur in the best 0.25 per cent. By combining SNP, protein, tissue expression, and genotypic frequency data, together with antigen presentation prediction algorithms, we propose a list of the best peptide candidates which could potentially induce the graft versus leukemia effect without causing graft versus host disease.

Availability: http://www.peptidecheck.org

Contact: Transfusionsmedizin@mh-hannover.de

1 INTRODUCTION

The role of minor histocompatibility antigens (mHags) in the context of hematopoietic stem cell transplantation is being intensely studied (Hambach and Goulmy, 2005). These antigens, which can potentially result from any polymorphic gene, have been implicated in causing the deadly graft versus host disease (GvHD) and present a hurdle for successful treatment of leukemia and other hematopoietic diseases following hematopoietic stem cell transplantation (HSCT) (Goulmy, et al., 1996). However, these immunological targets also prevent relapse when expressed on the surface of the patient's malignant cells (Spierings, et al., 2004). Here they are targeted by donor T cells, causing the so called graft versus leukemia effect (GvL). As we have reviewed previously, bioinformatics has become an important tool in investigating mHags (DeLuca and Blasczyk, 2007). We present here, a computation approach to predicting minor histocompatibility antigens, with special attention given to those antigens, which cause GvL. This system, named PeptideCheck, considers gene expression, polymorphism data, and antigen presentation prediction algorithms.

A given antigen can promote GvHD or GvL depending on its expression pattern across cell and tissue types. Because liver and epithelial cells are particularly affected by GvHD, it is logical that antigens which are expressed in these cells contribute to GvHD. On the other hand, antigens expressed exclusively in leukemia cells could have a targeted anti-tumor effect without causing GvHD. In fact, antigens specific to hematopoietic cells are also interesting targets for the GvL effect, as long as they only occur in the patient's original blood system, but not in the blood system of the donor after HSCT. This is the situation when hematopoietically expressed antigens are also mHags – i.e. they result from polymorphic mismatches between donor and recipient.

In principle, mHags can result from any genetic polymorphism which leads differential amino acid expression. In term of single nucleotide polymorphisms (SNP), examples include non-synonymous nucleotide replacements leading to an amino acid exchange, frame-shift causing nucleotide insertions or deletions, as well as mutations which either disrupt stop codons, or result in premature stop codons. The NCBI's dbSNP polymorphism database and the HapMap project are important resources for such data (Consortium, 2003; Smigielski, et al., 2000).

In addition to having to fulfill these genetic requirements, mHag candidates must be presented on the cell surface by the antigen presentation machinery (Rock and Goldberg, 1999). This process begins with proteasomal cleavage or proteins into peptide fragments. These peptides are then selectively loaded into MHC molecules by the transporter associated antigen processing (TAP) protein. Finally, the MHC-peptide complexes are carried to the cell surface whether they can interact with the T cell receptors found on the surface of T lymphocytes. Because each of these steps is selective and dependant on motifs found in the peptide sequences, it has been possible to develop algorithms for predicting the fate of peptide regions. Here, we employ the strategy of utilizing the processing scores to filter out a list of the most promising peptides. Finally the best candidates are those which have high processing scores for all applied algorithms, relevant SNP frequencies, and appropriate tissue-specific gene expression.

For the technique of integrating databases and algorithms to explore mHags, the state of the art includes systems such as SNEP (Schuler, et al., 2005) and SiPep (Halling-Brown, et al., 2006). SNEP extracts polymorphism data and sequences from SWISS-PROT (Boeckmann, et al., 2003) and calculates HLA binding using SYFPEITHI (Schuler, et al., 2007). SiPep utilizes dbSNP data, tissue expression data and combines proteasomal processing with HLA binding predictions. These systems however, are impractical for high throughput analysis. With PeptideCheck, we go several steps further to integrate user-defined gene expression analysis, and batch processing to analyze large amounts of user or public data conveniently.

2 METHODS

Prediction algorithms

The following prediction algorithms were applied to the peptide candidates: Proteasomal processing prediction by NetChop (Kesmir, et al., 2002), and the PepCleave predictor (Ginodi, et al., 2008). TAP binding by Peters et al. HLA binding prediction was performed with matrixes, modular matrices (DeLuca, et al., 2007),.

Data sources

SNP Data was imported from NCBI using the HTTP-based querying service, eUtilities. Only human non-synonymous coding SNPs were considered. The NCBI eFetch service was queried using the database dbSNP (Build 128), the format XML, and the TERM:

```
( Homo+sapiens [Organism])+AND+( snp+protein [Filter])+AND+
(         (         ( in+del [SnpClass]+OR+ mixed [SnpClass])+AND+
                    ( coding+nonsynonymous [Function+class]+OR+
                              reference [Function+class])
          )+
          OR+( coding+nonsynonymous [Function+class])+
)
```

eUtilities were also used to retrieve protein sequences from NCBI for proteins containing SNPs.

The genes from the dbSNP which were marked as coming from the Y chromosome were included and tagged as Y-linked..Futher Y-linked genes were fed into PeptideCheck by querying NCBI Entrez using the term: "y-linked"[title] AND (human[orgn]) and not ("pseudogene"[title]).

Generating peptides

Amino acid exchanges were made in the protein sequence. All possible peptides of length 15 containing both variants of the SNP were generated ands stored in an InterSystems Caché database. For immunoPaproc, 15 amino acids were required for the calculations, whereby the first 9th amino acid represents the C terminus. For immuneepitope database predictions, entire protein sequences were considered. For those SNPs which result in a frame shift or involved stop codons, peptides from the entire protein sequence were included. Such peptides were tagged as located “Before Mutation” (BM), “After Mutation” (AM) or “Containing Mutation” (CM) respectively.

SNP frequencies

The genotypic SNP frequency data provided by the dbSNP was supplemented with frequency data directly from the HapMap project (Consortium, 2003). The data is automatically downloaded from the online repository found at http://www.hapmap.org/downloads/frequencies/latest/rs_strand/non-redundant/, and then unzipped and stored. We chose to change the representation of genotypic frequency data to make it more practical in the context of allogeneic transplantation. We define PP frequency (presence of peptide) to be the sum of the homozygous and heterozygous frequencies of the individuals expression a peptide variant, and the AP frequency (absence of peptide) to be the frequency of individuals who are homozygous negative for the given peptide.

Expression

The cell and tissue expression data presented here were acquired from three sources. The first source is our own analysis of CML, CD34+, primary intestinal epithelial (PIE), normal human epidermal keratinocytes (NHEK) cells using GeneChip HG-U133A probe array (Affymetrix) with human renal proximal tubule cells (RPTEC) cells as control signal. The array contains a probe set for 22,283 oligonucleotide sequences and was utilized according to the manufacturer’s recommendations. RNA extracts from each cell type were processed to cDNA by reverse transcription, followed by in vitro transcription using biotinylated nucleoside triphosphates. After hypridization to the array and scanning, the results were interpreted using the MAS 5.0 software (Affymetrix).

A further source of expression data was the LeGeneD (Leukemia Gene Database http://www.bioinformatics.org/legend/) which are designated LEU in our system. The third source of data comes from GeneNotes (Shmueli, et al., 2003), and the cell types are listed as Bone Marrow and Liver in PeptideCheck.

GvL ligand ranking

Extracting GvL-relevant ligand candidates from all the peptides in the database involves a combination of filtering and ranking. Firstly, peptides are filtered by the criteria entered by the user – cell/tissue expression, antigen presentation prediction scores, SNP types and frequencies. The genes encoding list of filtered peptides are then ranked by the number of candidate antigens per gene. The resulting peptides can then be browsed gene for gene.

SNP validation

The validation of the SNPs was performed by PCR-sequencing-based typing sequencing (Horn, et al., 2006). The PCR products were subsequently sequenced in both forward and reverse directions by a cycle sequencing kit (Big Dye terminator, Applied Biosystems, Foster city, CA) using an Applied Biosystems 3730 sequencer and the data were analyzed by the SeqMan II program version 5.7 (GATC, Konstanz, Germany). Because each SNP was associated with a particular HLA ligand, only samples which were positive for the correct HLA allele were tested for the given SNP.

3 RESULTS

Quantity and quality of data

In total, 48,905 SNP entries were imported from the dbSNP. These SNPs are found within 15,898 genes – roughly half of the estimated number of genes in humans. Because genes can be associated with multiple protein sequences at NCBI, 23,798 proteins were imported. The total number of unique peptide sequences contained in the system is 1,854,676. These and other statistics can be found in Table 1.

Table 1 – Number of entries database tables.

Data from dbSNP build 128	
Total SNPs	48,905
SNPs causing AA exchange	48,194
SNPs causing stop codon [a]	6,918
SNPs removing stop codon [a]	469
Single nucleotide insertions	2,936
Single nucleotide deletions	3,410
Total validated SNPs	22,918
SNPs validated by HapMap [b]	19,046
SNPs validated by Frequency [b]	17,532
Additional entries	
Total Genes	15,898
Total Proteins	23,798
Total Peptides	1,854,676

a. SNPs involving stop codons do not include frameshift mutations.
b. Validations as given in the dbSNP.

It should be noted that the ratios of proteins to genes in the system simply reflects the way that protein and gene data are reported to NCBI, and do not necessarily reflect biological events (e.g. mutations or alternative splicing, etc.).

Three sources of gene expression data have been included so far. Our own Affymetrix analysis resulted in 2853 CML expressed, 2714 CD34 expressed, 1953 PIE, 2833 NHEK, 2960 RPTEC, . Furthermore, 48 leukemia expressed genes were included. Additionally, 5514 bone marrow and 5575 liver genes were included. Finally, 12 Y chromosome associated genes were included. It should be noted that any user may upload any additional gene expression data.

Table 2 –Quantity of data from known mHags reflected in the PeptideCheck database

dbMinor					PeptideCheck				
Name	**Gene**	**Sequence [a]**	**HLA**	**SNP [b]**	**Validated**	**Protein**	**Expression**	**Cleavage [c]**	**HLA [d]**
HA-1	HMHA1	VL***H***DDLLEA	A*02	rs1801284 ✓	Yes	NP_036424	-	-1.80	-23.02
HA-1/B60	HMHA1	KECVL***H***DDL	B*60	rs1801284 ✓	Yes	NP_036424	-	-11.88	
HA-2	EAL23748	YIGEVLVS***V***	A*02	-	-	-	-	-4.49	-20.21
HA-3	AKAP13	V***T***EPGTAQY	A*01	rs7162168 (rs2061821)	Yes	NP_006729	Bone Marrow, Liver	-2.77	-19.57
HA-8	KIAA0020	***R***TLDKVLEV	A*02	rs2173904	Yes	NP_055693	Bone Marrow, CD34, CML, NHEK, RPTEC	-2.34	-21.17
HB-1H	HMHB1	EEKRGSL***H***VW	B*44	rs57824 (rs161557)	Yes	NP_067005	RPTEC	-1.71	
HB-1Y	HMHB1	EEKRGSL***Y***VW	B*44	rs57824 (rs161557)	Yes	NP_067005	RPTEC	-1.71	
ACC-1	BCL2A1	DYLQ***Y***VLQI	A*24	rs1138357 ✓	Yes	NP_0004040	Bone Marrow, CD34, CML, RPTEC	-10.13	-22.65
ACC-2	BCL2A1	KEFED***D***IINW	B*44	rs3826007 ✓	Yes	NP_0004040		-2.32	
SP110 (HwA-9)	SP110	SLP***R***GTSTPK	A*03	rs1365776	-	-	-	-10.00	
PANE1 (HwA-10)	CENPM	***R***VWDLPGVLK	A*03	-	-	-	-	-4.05	
UGT2B17/A29	UGT2B17	AELLNIPFLY	A*29	Gene deletion	-	-	-	-2.34	
UGT2B17/B44	UGT2B17	AELLNIPFLY	B*44	Gene deletion	-	-	-	-2.34	
LRH-1	P2RX5	TPNQRQNVC	B*07	rs5818907 ✓	No	NP_002552	Bone Marrow, Liver	-1.36	-27.91
LB-ECGF-1H	TYMP	RP***H***AIRRPLAL	B*07	-	-	-		-6.20	
CTSH/A31	CTSH	ATLPLLCA***R***	A*31	rs2289702 ✓	Yes	NP_004381	Bone Marrow, CML, PIE, Liver, NHEK, RPTEC	-7.37	-22.58
CTSH/A33	CTSH	WATLPLLCA***R***	A*31	rs2289702 ✓	Yes	Pept. Too long		-1.88	
LB-ADIR-1F	TOR3A	SVAPALAL***F***PA	A*02	rs2296377	-	-	-	-0.20	
ACC-6	HMSD	MEIFIEVFSHF	B*44	Exon deletion	-	-	-	1.19	
A1/HY	USP9Y	IVD***C***LTEMY	A*01	Y-linked	-	NP_004645	Y-linked, Bone Marrow, CD34, CML, Liver, RPTEC	-4.19	-21.09
A2/HY	JARID1D	FIDSYICQV	A*02	Y-linked	-	NP_004644	Bone Marrow, CML, Liver, RPTEC	-9.10	-24.47
B7/HY	JARID1D	SPSVDKARAEL	B*07	Y-linked	-	NP_004644		0.25	
A33/HY	TMSB4X	EVLLRPGLHFR	A*33	Y-linked	-	-	-	-0.45	
B8/HY	UTY	LPHNHTDL	B*08	Y-linked	-	NP_009056	Y-linked, Bone Marrow, CD34, CML, RPTEC	3.62	
B60/HY	UTY	RESEEESVSL	B*60	Y-linked	-	NP_009056		1.24	
B52/HY	RPS4Y1	TIRYPDPVI	B*52	Y-linked	-	NP_000999	Bone Marrow, Liver	-5.81	-42.40
DRB3*0301/HY	RPS4Y1	VIKVNDTVQI	DRB3*03	Y-linked	-	NP_000999	Bone Marrow, Liver	2.13	
DRB1*1501/HY	DDX3Y	SKGRYIPPHLR	DRB1*15	Y-linked	-	NP_004651	Bone Marrow, Liver	-3.93	
DQ5/HY	DDX3Y	HIENFSDIDMGE	DQB1*05	Y-linked	-	NP_004651	Bone Marrow, Liver	-3.48	

This table reflects information on the known mHags as reported in the dbSNP (left side) and in PeptideCheck (right side) to illustrate the extent to which PeptideCheck can reflect real mHags. **a.** The sequence give is the immunogenetic peptide. The polymorphic residue is given in ***bold italics***. **b.** SNP entries given with a ✓ were identical between dbSNP and PeptideCheck. In the cases where PeptideCheck uses a different, but equally correct rs entry, it is given in parenthesis. "Y-linked" does not refer to dbSNP data, but simply that the allogenicity resulting from gender difference. **c.** Proteasomal processing score by PepCleave. Higher values are better and scores over -4 can be considered good. **d.** Matrix-based HLA binding score for the mHag associated allele. Higher values are better, and scores greater than -27 are quite good.

Coverage of known mHags

Because the goal of this system is to identify possible mHag candidates, it is interesting to investigate whether known mHag peptides are found within the database. The sequences and reference data of known mHags was downloaded from the Minor Histocompatibility Knowledge Database (dbMinor) hosted at the Leiden University Medical Center website (Spierings, et al., 2006). We searched for these peptide sequences in PeptideCheck to determine whether the data correlate to the published data (Table 2).

There were 29 minors listed in dbMinor which result from 21 unique polymorphisms. All 14 reported coding non-synonymous SNPs from dbMinor were also reflected within the PeptideCheck database. Of the 7 missing SNPs, 2 were simply not reported to the dbSNP (HA-2 and LB-ADIR-1F), excluding them from our system as a result. Because we considered only coding non-synonymous SNPs, it was expected that polymorphisms such as alternative splicing, gene deletion, etc. would not be encompassed in the model. The 5 remaining missing SNPs can be attributed to this kind of model limitation. The mHag encoded by SP110 results from transpeptidation (Warren, et al., 2006). The mHag encoded by CENPM results from alternative transcription lead to the incorporation of an additional exon (Brickner, et al., 2006).The two mHags encoded by UGT2B17 are caused by gene deletion (Murata, et al., 2003). Furthermore, alternative splicing, causing an exon deletion in HMSD causes the mHag, ACC-6 (Kawase, et al., 2007). The minor A33/HY from the gene TMSB4X was not found because it is encoded by an unconventional open reading frame (Torikai, et al., 2004). The same applies to LB-ADIR-1F. (van Bergen, et al., 2007)

In several cases, the dbSNP entries used by PeptideCheck to generate known mHags differed from those of the dbMinor, revealing outdated or erroneous SNPs in dbMinor. For HA-3 the originally reported dbSNP entry rs7162168 claims an amino acid exchange at position 1216 in the sequence found in NP_006729. However the actual position of the exchange is 452, which is given in a different dbSNP entry retrieved by our system: rs2061821. For HB-1H and HB-1Y, the reported rs57824 was outdated, but our system identified the correct replacement: rs161557.

Table 3 – Ranked mHag candidates

HLA-A*0101 binders									
Rank	**Gene**	**Peptide**	**Clea-vage**	**HLA**	**Pep. Pos.**	**AA Pos.**	**dbSNP**	**Alt. res.**	**Expression**
1	MMP8	S***S***DPGALMY	2.17	-15.39	228	229	rs12792229	Y	Bone Marrow
2	AKAP13	V***T***EPGTAQY (**HA-3**)	-2.35	-15.85	451	452	rs2061821	M	Bone Marrow, Liver
3	TACSTD2	***E***VDIGDAAY	-1.21	-16.83	216	216	rs2061821	D	Bone Marrow
4	TJP3	ETDGEG***D***AY	-0.21	-17.05	841	847	rs10408494	G	-
5	TPO	***V***ADKILDLY	2.36	-17.08	445	445	rs10189135	M	-
6	HIRIP3	EAAPP***G***ELY	2.11	-17.10	516	521	rs11643314	W	Bone Marrow
7	MGC35048	LTEEEAA***L***Y	0.02	-17.19	213	220	rs7191155	P	-
8	ASPSCR1	AA***D***VLVARY	-1.59	-17.31	485	487	rs13087	E	Bone Marrow, Liver
9	COG3	V***S***DLAATAY	-0.91	-17.62	746	747	rs2274285	N	Bone Marrow, Liver
10	GPR153	A***T***ELLANQY	-2.95	-17.64	324	325	rs13374337	I	Bone Marrow, Liver
HLA-A*0201 binders									
Rank	**Gene**	**Peptide**	**Clea-vage**	**HLA**	**Pep. Pos.**	**AA Pos.**	**dbSNP**	**Alt. res.**	**Expression**
1	DEFA4	ALL***P***AILLV	-2.91	-16.72	5	8	rs28661751	A	Bone Marrow,CD34,CML
2	PTGS1	LL***L***PLPVLL	-2.61	-17.26	15	17	rs3842787	P	CD34, CML, NHEK, PIE, RPTEC
3	MC1R	LLLEA***S***ALV	-2.56	-17.29	99	104	rs2229617	G	-
4	IGFBP7	LL***L***GAAGLL	-2.88	-17.96	9	11	rs11573021	F	Bone Marrow, CD34, CML, Liver, NHEK, RPTEC
5	ABCB8	***I***LALGAALV	-2.27	-18.13	136	136	rs4148844	V	Bone Marrow, Liver
6	WFS1	ILVA***G***LALV	-1.96	-18.43	572	576	rs1805069	S	Liver, NHEK, RPTEC
7	CPZ	***L***LLLLTVLV	-2.92	-18.47	6	6	rs2302583	P	Bone Marrow, Liver
8	SH3RF2	AL***A***KATTLV	-2.86	-18.51	708	710	rs1056149	G	-
9	PKDREJ	LLLILI***V***LL	-2.89	-18.54	1723	1729	rs9626829	I	-
10	FLJ14346	ALGG***A***LALA	0.12	-18.54	114	118	rs1043059	V	-

A total of 822,299 peptides were filtered with the requirements of being coded by a validated SNP and having a PepCleave score > -3.0. Peptides were then ranked by their matrix-based HLA binding score. The known mHag, HA-3 can be found at rank 2 under the A*0101 binders. The polymorphic position in the peptide is given in ***italic bold***. **Cleavage** refers to PepCleave scores. **HLA** refers to matrix-based HLA binding scores. **Pep. Pos** is the position of the peptide within the protein sequence. **AA pos.** is the position of the polymorphic residue within the original protein sequence. **Alt. res.** Is the alternate residue given by the dbSNP entry.

Ranking of ligand candidates

To find mHag candidates, the peptides were filtered and ranked. As filtering criteria, the peptides were required to be encoded by a validated missense SNP and have a PepCleave Proteasomal processing score above -3.0. The peptides were then ranked by according to their HLA binding scores. The best 10 candidates for A*0101 and A*0201 are given in Table 3.

As a result of this ranking, several of the known mHags could be reproduced. The total number of peptides considered (those resulting from missense SNPs) was 822,299. The mHag, HA-3 was ranked at place 2 binding to HLA*0201. The HLA*0101 binding known mHags HA-8 and HA-1 were found at places 330 and 1,748 respectively. To put these numbers in relation, it should be noted that even place 1,748 is within the top quarter of the top one per cent of the peptides considered.

The major motivation for creating PeptideCheck was to help identify GvL-relevant ligand candidates. GvL-relevance is determined by a cell expression which is specific to hematopoietic tissues, and not present in tissues at risk to GvHD. For Proteasomal processing, the PepCleave score of -3.0 was used. This score was determined by the orientation provided by Table 2. A HLA binding score of -20 was applied, which is associated with a specificity of over 99% (DeLuca, et al., 2007) for HLA-A*0201. Further requirements include both PP and AP frequencies of at least 10%. These filters resulted in the 13 GvL-relevant ligand candidates listed in Table 4.

Table 4 – GvL relevant ligand candidates binding HLA-A*0201

Gene	Peptide	Clea-vage	HLA	Pep. Pos.	AA Pos.	dbSNP	Alt. Res.	Frequency		Expression
								PP	AP	
LY64	LLAILLF***L***A	-0.19	-19.91	641	648	rs2230524		10	90	CML
LY64	LLF***L***AVKYL	-3.68	-21.61	645	648	rs2230524		10	90	CML
SLC4A1	SLALPF***I***LI	-3.17	-21.23	856	862	rs5026		15	84	Bone Marrow
SLC4A1	LALPF***I***LIL	-4.00	-21.18	856	862	rs5026		15	84	Bone Marrow
EGFL6	IAV***N***GVLLV	1.88	-21.98	532	535	rs16979033	D	23	77	CML
EGFL6	IAV***D***GVLLV	1.12	-20.98	532	535	rs16979033	N	88	12	CML
RIF1	AT***V***ENAVLL	-1.22	-21.62	1360	1362	rs2123465		65	35	Bone Marrow
RIF1	L***L***AQISALA	-0.14	-20.85	2417	2418	rs1065177		63	67	Bone Marrow
MMP8	***S***LKTLPFLL	-3.51	-20.22	3	3	rs17099450		18	82	Bone Marrow
CARD8	***YLVPSDALL***	-2.45	-20.82	229	10	rs2043211	stop	38	62	Bone Marrow
STARD9	***I***LPGALTRV	-3.85	-20.14	2860	2860	rs8031218		82	18	Bone Marrow
FLJ21144	GE***G***EGVLLV	-2.96	-21.90	170	172	rs11208299		52	48	CD34, CML
LOC401115	CLPAASAA***V***	-3.80	-21.90	123	131	rs10003030		85	15	Bone Marrow

The polymorphic position in the peptide is given in ***italic bold***. **Cleavage** refers to PepCleave scores. **HLA** refers to matrix-based HLA binding scores. **Pep. Pos** is the position of the peptide within the protein sequence. **AA pos.** is the position of the polymorphic residue within the original protein sequence. **Alt. res.** is the alternate residue given by the dbSNP entry. The SNP for CARD8 results in a premature stop codon and the reported peptide is downstream from this mutation.

SNP typing results

To determine whether the SNPs of candidate peptides occur with clinically relevant frequencies, sequencing-based SNP typing was performed on from healthy blood donors. The genes were chosen based upon the number of associated SNPs which lead to peptides with high prediction scores. Sequencing-based typing was performed for a set of SNPs which were reported in the database as validated either by HapMap or by frequency data from the dbSNP. Additionally a set of SNPs with no validation data were typed. In both cases, only those donors were typed for a given SNP when they were previously shown to carry the HLA allele predicted to bind the SNP-derived peptide. A selection of SNP typings are shown in Table 5. Of the 6 previously validated SNPs, all but one were confirmed. None of the non-validated SNPs could be found after 8 to 30 typings.

Table 5 – Confirmation of SNPs

Gene	SNP	Validated	Ref. Residue		Alt. Residue		Heteroz.	Total	Confirm.
			Res.	N	Res.	N			
EGFL6	rs16979033	Yes	D	17	N	0	0	17	No
F13A1	rs5982	Yes	P	6	L	1	1	7	Yes
TBXAS1	rs4526	Yes	A	8	T	0	0	8	No
ZNF117	rs3807069	Yes	Y	0	C	4	4	8	Yes
BTN3A1	rs4712990	Yes	T	1	P	5	2	8	Yes
TREM1	rs2234237	Yes	T	41	S	0	7	48	Yes
BRCA2	rs11426352	No	Del	30	-	0	0	30	No
EGF	rs11569144	No	Del	13	-	0	0	13	No
TBXAS1	rs2286199	No	Del	8	-	0	0	8	No
ITGA9	rs5848136	No	Del	12	-	0	0	12	No
CD86	rs10703820	No	Del	8	-	0	0	8	No

Validated = listed as validated by the dbSNP or HapMap. **Ref. Residue** = reference residue; **Alt. Residue** = alternative residue. **N** = number of positive typings. **Heteroz.** = number of samples typed to be heterozygous. **Confirm**. = conclusion of whether or not the SNP could be independently confirmed in this study.

Web interfaces

PeptideCheck is freely available online at http://www.peptidecheck.org.

There are 3 ways to query data from PeptideCheck:

1. **Search with Peptides**: user enters peptide sequence to retrieve gene, SNP data and antigen presentation scores
2. **Search with Genes**: user enters gene symbol, and receives all SNPs and peptide data for this gene
3. **GvL Ligand Finder**: user enters expression, SNP, and prediction score requirements and receives a ranked list of GvL-relevant ligand candidates

Search with Peptides

The Search with Peptides function is particularly useful for those who have experimentally eluted and sequence peptides, and wish to find information about these peptides such as protein origin, SNP data, expression data, and presentation prediction scores. These data can be entered as a single peptide, or a long list of peptides. The peptides can be filtered by making appropriate prediction score requirements, as well as requiring which types of SNPs should be included. The result is a table of the peptides which match the sequence and other criteria together with gene, SNP, expression and presentation score data.

Search with Genes

If the user is interested in peptide presentation and polymorphism data for one or more specific genes, than they may use the Search with Genes option. A list of gene symbols can be given and processing score filtering can be optionally included. The resulting data is similar to that when searching with Peptides.

GvL Ligand Finder

The goal is to find peptide which can be effective mHags. Source genes are narrowed down by choosing an expression profile. This step may be excluded. Next, the user makes decision about SNP criteria: these include the genotype frequency thresholds, type of SNPs (ins, del, missense, etc). Finally the user chooses which prediction algorithms to consider, and which threshold to use. Default thresholds are suggested by the system. After entering the filtering criteria, the user is presented with a ranked list of peptide candidates. Peptides are ranked by the number of peptide candidates found for a given gene. The information displayed includes gene symbol, expression profile, chromosome, total ligands (per gene), peptide sequence, prediction scores, peptide position (position of peptide in original protein sequence), amino acid position (position of SNP in original protein sequence), PP/AP(Presence of Peptide frequency, Absence of Peptide frequency), SNP type (missense, insertion, etc), dbSNP rs Link, validation (hap map or by frequency), local position (position of exchange within the peptide), and the protein accession number with link to NCBI.

It is also possible for users to upload new tissue or cell type expression data. They can do this by uploaded an Affymetrix result file for automatic interpretation. Alternatively, then can simply upload a list of genes and define with which cell types they are associated. The resulting expression data is then active when the user searches for peptides.

4 DISCUSSION

By providing this compilation of databases and algorithms online at www.peptidecheck.org, we hope to offer the mHag community a resource which can offer practical assistance in discovering and analyzing GvL-relevant peptides. This is the first system offering combined antigen presentation prediction algorithms for mHag analysis and in a manner convenient for high throughput investigation of sequences from experimentally eluted peptides.

The novel representation of SNP frequency as PP (presence of peptide) and AP (absence of peptide) frequencies is specific to the situation of allogeneic transplantation, and is practical for quickly determining the clinic relevance of SNP data. Our own SNP typing confirmation demonstrates the futility of searching for non-validated SNPs. This underscores the value of the HapMap frequency data, which is reflected in the PP and AP scores in the system. When searching for peptides, the PeptideCheck users can conveniently provide thresholds for these values.

The comparison of the data in this system to that of the dbMinor, which contains data on known mHags, has helped to clarify the extent to which bioinformatic systems can simulate immunogenetic processes. Clearly coding non-synonymous mutations leading to a single amino acid exchange are well suited to be reproduced using computer algorithms. The situation becomes more complicated when frame shifts, splice cites, or promoter regions are involved. Despite this, PeptideCheck does also incorporate insertion / deletion SNPs. Here we have chosen to generate peptides from the full length of the protein sequence, and to designate them as occurring before or after the beginning of the frame shift, or as containing the SNP cite. Most likely, peptides occurring after the frame shift are most likely to be immunogenetic. Since there is no data on this, we simply choose to label the peptides accordingly, and allow the PeptideCheck user to decide.

The matrix-based HLA binding prediction algorithm produced strikingly high scores for known mHags. Ranking our lists of peptide candidates based upon these scores proved to be very successful, if success is measured by the presence of known mHags near the top of the list.

The greatest limitation of this system currently is the fact that HLA binding scores are only made for peptides having nine amino acids. However, this analysis was necessary to determine if there is merit in the computational approach to mHag identification. This being the case, we will expand the system to include peptides of different lengths. Furthermore, a database of gene deletion frequencies would greatly augment this system.

An important advantage of this automated approach is that it is adaptable to potential forms of individualized medicine. As the price of SNP microarrays decreases, the ability for large scale SNP typing for an individual patient and donor pair becomes reality. Using this input to generate GvL-relevant ligand candidates which could be then synthesized and utilized for ex vivo T-cell stimulation during adoptive transfer offers great potential.

ACKNOWLEDGEMENTS

This work was financed by the Federal Ministry of Education and Research, Germany, funding number 01DS001C; the authors are solely responsible for content. The authors are very appreciative of the assistance of Yoram Louzoun of Bar Ilan University, Israel. The implementation of PeptideCheck would not have been possible in its current form without the technology provided by the InterSystems Caché database, for which InterSystems provided sponsored licensing. In particular, Michael Braam of InterSystems provided generous technical support.

REFERENCES

Boeckmann, B., Bairoch, A., Apweiler, R., Blatter, M.C., Estreicher, A., Gasteiger, E., Martin, M.J., Michoud, K., O'Donovan, C., Phan, I., Pilbout, S. and Schneider, M. (2003) The SWISS-PROT protein knowledgebase and its supplement TrEMBL in 2003, *Nucleic Acids Res*, **31**, 365-370.

Brickner, A.G., Evans, A.M., Mito, J.K., Xuereb, S.M., Feng, X., Nishida, T., Fairfull, L., Ferrell, R.E., Foon, K.A., Hunt, D.F., Shabanowitz, J., Engelhard, V.H., Riddell, S.R. and Warren, E.H. (2006) The PANE1 gene encodes a novel human minor histocompatibility antigen that is selectively expressed in B-lymphoid cells and B-CLL, *Blood*, **107**, 3779-3786.

Consortium, T.I.H. (2003) The International HapMap Project, *Nature*, **426**, 789-796.

DeLuca, D.S. and Blasczyk, R. (2007) The immunoinformatics of cancer immunotherapy, *Tissue Antigens*, **70**, 265-271.

DeLuca, D.S., Khattab, B. and Blasczyk, R. (2007) A modular concept of HLA for comprehensive peptide binding prediction, *Immunogenetics*, **59**, 25-35.

Ginodi, I., Vider-Shalit, T., Tsaban, L. and Louzoun, Y. (2008) Precise score for the prediction of peptides cleaved by the proteasome, *Bioinformatics*, **24**, 477-483.

Goulmy, E., Schipper, R., Pool, J., Blokland, E., Falkenburg, J.H., Vossen, J., Gratwohl, A., Vogelsang, G.B., van Houwelingen, H.C. and van Rood, J.J. (1996) Mismatches of minor histocompatibility antigens between HLA-identical donors and recipients and the development of graft-versus-host disease after bone marrow transplantation, *N Engl J Med*, **334**, 281-285.

Halling-Brown, M., Quartey-Papafio, R., Travers, P.J. and Moss, D.S. (2006) SiPep: a system for the prediction of tissue-specific minor histocompatibility antigens, *Int J Immunogenet*, **33**, 289-295.

Hambach, L. and Goulmy, E. (2005) Immunotherapy of cancer through targeting of minor histocompatibility antigens, *Curr Opin Immunol*, **17**, 202-210.

Horn, P.A., Verboom, M. and Blasczyk, R. (2006) HLA-A*2313 is closest to A*2301 but is likely to stimulate T cells when mismatched, *Tissue Antigens*, **67**, 166-167.

Kawase, T., Akatsuka, Y., Torikai, H., Morishima, S., Oka, A., Tsujimura, A., Miyazaki, M., Tsujimura, K., Miyamura, K., Ogawa, S., Inoko, H., Morishima, Y., Kodera, Y., Kuzushima, K. and Takahashi, T. (2007) Alternative splicing due to an intronic SNP in HMSD generates a novel minor histocompatibility antigen, *Blood*, **110**, 1055-1063.

Kesmir, C., Nussbaum, A.K., Schild, H., Detours, V. and Brunak, S. (2002) Prediction of proteasome cleavage motifs by neural networks, *Protein Eng*, **15**, 287-296.

Murata, M., Warren, E.H. and Riddell, S.R. (2003) A human minor histocompatibility antigen resulting from differential expression due to a gene deletion, *J Exp Med*, **197**, 1279-1289.

Rock, K.L. and Goldberg, A.L. (1999) Degradation of cell proteins and the generation of MHC class I-presented peptides, *Annu Rev Immunol*, **17**, 739-779.

Schuler, M.M., Donnes, P., Nastke, M.D., Kohlbacher, O., Rammensee, H.G. and Stevanovic, S. (2005) SNEP: SNP-derived epitope prediction program for minor H antigens, *Immunogenetics*, **57**, 816-820.

Schuler, M.M., Nastke, M.D. and Stevanovikc, S. (2007) SYFPEITHI: database for searching and T-cell epitope prediction, *Methods Mol Biol*, **409**, 75-93.

Shmueli, O., Horn-Saban, S., Chalifa-Caspi, V., Shmoish, M., Ophir, R., Benjamin-Rodrig, H., Safran, M., Domany, E. and Lancet, D. (2003) GeneNote: whole genome expression profiles in normal human tissues, *C R Biol*, **326**, 1067-1072.

Smigielski, E.M., Sirotkin, K., Ward, M. and Sherry, S.T. (2000) dbSNP: a database of single nucleotide polymorphisms, *Nucleic Acids Res*, **28**, 352-355.

Spierings, E., Drabbels, J., Hendriks, M., Pool, J., Spruyt-Gerritse, M., Claas, F. and Goulmy, E. (2006) A uniform genomic minor histocompatibility antigen typing methodology and database designed to facilitate clinical applications, *PLoS ONE*, **1**, e42.

Spierings, E., Wieles, B. and Goulmy, E. (2004) Minor histocompatibility antigens--big in tumour therapy, *Trends Immunol*, **25**, 56-60.

Torikai, H., Akatsuka, Y., Miyazaki, M., Warren, E.H., 3rd, Oba, T., Tsujimura, K., Motoyoshi, K., Morishima, Y., Kodera, Y., Kuzushima, K. and Takahashi, T. (2004) A novel HLA-A*3303-restricted minor histocompatibility antigen encoded by an unconventional open reading frame of human TMSB4Y gene, *J Immunol*, **173**, 7046-7054.

van Bergen, C.A., Kester, M.G., Jedema, I., Heemskerk, M.H., van Luxemburg-Heijs, S.A., Kloosterboer, F.M., Marijt, W.A., de Ru, A.H., Schaafsma, M.R., Willemze, R., van Veelen, P.A. and Falkenburg, J.H. (2007) Multiple myeloma-reactive T cells recognize an activation-induced minor histocompatibility antigen encoded by the ATP-dependent interferon-responsive (ADIR) gene, *Blood*, **109**, 4089-4096.

Warren, E.H., Vigneron, N.J., Gavin, M.A., Coulie, P.G., Stroobant, V., Dalet, A., Tykodi, S.S., Xuereb, S.M., Mito, J.K., Riddell, S.R. and Van den Eynde, B.J. (2006) An antigen produced by splicing of noncontiguous peptides in the reverse order, *Science*, **313**, 1444-1447.

10 The MHC Ontology

Title

Development and immunoinformatic application of the MHC ontology

Authors

DeLuca, D.S., Beisswanger, E., Wermter, J., Horn, P.A., Hahn, U., Blasczyk, R.

Prepared for

Bioinformatics, 2008

Development and immunoinformatic application of the MHC ontology

David S. DeLuca[1], Elena Beisswanger [2], Joachim Wermter[2], Peter A. Horn[1], Udo Hahn[2], Rainer Blasczyk[1]

[1] Institute for Transfusion Medicine, Hannover Medical School, Carl-Neuberg-Str. 1, Hanover
[2] Institut für Germanistische Sprachwissenschaft, Friedrich-Schiller-Universität, Fürstengraben 30 Jena

Abstract

Biomedical ontologies have become an increasingly important tool in bioinformatics. In the field of immunogenetics, the complications caused by extreme levels of polymorphism are being met by bioinformatic software and databases. Here we apply the principles of ontology development to the challenges presented by computationally representing alleles of the major histocompatibility complex, resulting in the MHC Ontology. Importantly for human immunogenetics, a detailed level of HLA classification is included. We demonstrate the utility of this ontology in several bioinformatic applications. Currently the MHC Ontology serves as a database schema, supports user and software interactions, and the semantic annotation of scientific documents. Through continual collaboration, this ontology has the potential to serve as a basis for data interoperability between centralized immunogenetic databases, and the applications used by researchers and laboratories.

Availability: The MHC Ontology is available via the BioPortal: http://www.bioontology.org/tools/portal/bioportal.html, and at: http://purl.org/stemnet/

Use of type face

Ontology classes are written in **bold**

Relations are written in *italics*

Annotation elements are written in ***bold italics***

Introduction

Emergence and relevance of ontologies in biology

In computer science, the concept of ontology refers to a set of terms within a knowledge domain which are linked together and organized based upon their relation to each other. The most important relation is the *is a* relation which allows a child term to inherit all of the characteristics of its parent. This fundamental approach has been adopted by biologists in recent years to help formalize scientific terminology in complicated biological domains such as genetics, cell taxonomy, and protein taxonomy. The Open Biomedical Ontologies (OBO) library (http://www.bioontology.org/repositories.html#obo) is an umbrella resource for coordinating ontology projects (Smith, et al., 2007). In addition to the benefits of providing a defined set of vocabulary for the scientific community, the remarkable achievement of ontology development lies in organizing real world concepts into a format which is easily assessable to computer systems. This computability makes ontologies a relevant and important concept in bioinformatics.

Challenges of human immunogenetics

Many of the challenges posed by the (human) major histocompatibility complex to researchers of pathology, transplantation, and immunology result from the high level of polymorphism in this area of chromosome six. In particular, genetic typing of human leukocyte antigen (HLA) alleles to ensure patient/donor compatibility following hematopoietic stem cell transplantation relies heavily on computer systems for data storage, organization and interpretation. HLA typing can be performed with various levels of precision (Little, 2007). This leads to a hierarchical categorization of HLA alleles as depicted in Table I. The classic method of HLA tying is via serological testing. This method is the least precise, because many HLA proteins share the structural domains which are recognized by the antibodies during typing. Serological typing provides the serological group of an HLA protein. Genetics-based HLA typing can determine which HLA allele is present in a more specific manner, but is also performed with various levels of precision, resulting in so-called high, low, or medium resolution results.

Table I. Hierarchical nature of HLA alleles and nomenclature under the example A*0201

Serological Groups [a]	***A2***-reactive Proteins [b]	Alleles coding for ***A*0201*** [c]
A1	***A*0201*** →	A*02010101
A2 →	A*0202	A*02010102L
A203	A*0203	A*020102
A210	A*0204	A*020103
A3	A*0205	A*020104
A9	A*0206	A*020105
A10	A*0207	A*020106
A11	A*0208	A*020107
A19	A*0209	A*020108
A2403	A*0210	A*020109
A28	A*0211	A*020110
A36	A*0212	A*020111
A43	A*0213	A*020112
A80	A*0214	A*020113
	A*0216	A*020114
	A*0217	A*020115
	etc	A*020116
		A*020117

a. Here the serological splits are not listed. A full list of serological groups and splits can be found at the Anthony Nolan HLA Informatics Group Website. **b**. These proteins were defined to be A2 reactive in the HLA Dictionary(Schreuder, et al., 2005). **c**. The known alleles encoding the same proteins as the reference allele A*02010101 as of IMGT/HLA Database Release 2.20.0, 11 January 2008.

Determination of the serological group (e.g. A2) as well as “two-digit” genetics-based typing results (e.g. A*02) are considered low resolution typing. Serological groups have subclasses known as “serological splits”. Splits are subsets of HLA antigens which are reactive with antibodies of higher specificity, making for a more precise determination. Donor and recipient alleles are considered to be a low resolution match when each allele has been determined to belong to the same serological or two digit group. Therefore, the definition of allelic groups is highly clinically relevant when dealing with HLA alleles. The HLA nomenclature has been designed to make the group of an allele immediately apparent by noting the group in the first two digits of the allele name (Marsh, 2003). Recently, as the number of determined HLA alleles has increased, the third and fourth digits of the HLA allele have been proven to be inadequate. Each two digit group can contain 100 four digit groups. This has resulted in the A*02 group “spilling over” into the A*92 group(Marsh, et al., 2002). While those who have years of experience with the HLA system know that an A*92 allele is actually an A*02 allele, this is likely to create much confusion for the next generation of researches and clinicians. Currently, the only further example of a group extension is B*15 spilling over into B*95. Computer systems dealing with the HLA system must also consider these exceptions. The creation of an ontology is a good opportunity to formally define allelic grouping for use by people and computers.

Text mining for scientific knowledge and document retrieval

An important motivation for creating an MHC ontology is for its application in text mining as part of the StemNet project (http://www.stemnet.de/)(Hahn, et al., 2007). The aim of StemNet is to create a knowledge resource which combined biological databases with unstructured biological literature. The mapping of concepts in natural language texts to structured database entries requires an automated annotation process. Here, the MHC ontology provides an annotation vocabulary, hierarchical organization of concepts, as well as the importation relations between terms.

Methods

Construction and implementation

The MHC ontology is represented in OWL DL, a sublanguage of the Web Ontology Language (OWL) (http://www.w3.org/TR/owl-features/), which is an extension of the Resource Description Framework (RDF) (http://www.w3.org/RDF/). Basically the MHC ontology is a directed acyclic graph (DAG) based on *rdfs:subClassOf* relations between classes and their parent classes. In the ontology domain specific terms are represented in terms of OWL classes with a Unified Resource Identifier (URI). Class URIs consist of a general namespace and the name of the class within that namespace (called local name). Because of character restrictions within URIs, we used the RDF property ***rdfs:label*** to provide an unformatted, human-readable version of the class name in addition to the class URI.

Metadata describing the resource MHC ontology as such are provided using Dublin Core Metadata Initiative (DCMI) Metadata terms (http://dublincore.org/) such as ***dc:creator***, ***dc:date***, ***dc:subject*** and ***dc:title***. The term ***dc:source*** is use as holder for literature or database references and is used throughout the ontology. Furthermore, we defined the custom OWL annotation properties ***definition***, ***synonym***, and ***reference*** to provide textual definitions of classes, exact synonyms of class names, and references to similar terms in other terminologies.

While the upper level of the MHC Ontology was created manually using the Stanford University ontology editor, Protégé (http://protege.stanford.edu), subclasses of the **Human_MHC_Allele** class and its subclasses were automatically generated extracting terms and taxonomic relations from external databases. These terms have been put into an external OWL file we call the HLA Ontology. The HLA Ontology is included in the MHC Ontology using an ***owl:imports*** statement. The generation algorithm for the HLA Ontology was written in Java. Extracted data were exported into the OWL format using the Jena 2 Ontology API (Carroll, et al., 2004). An additional version of the HLA Ontology was created in a novel XML format, specific to the HLA system. These versions of the ontology were exported in XML using the JDOM API (http://www.jdom.org/).

Sources of data

To ensure compatibility with the greater ontology community, top-level classes, such as allele and gene were created in equivalence with terms in the Sequence Ontology (SO)(Eilbeck, et al., 2005). Subclasses for the **Canine_MHC_Allele** class were provided by the DLA Nomenclature Reports, as provided by the Immuno Polymorphism Database (IPD) (http://www.ebi.ac.uk/ipd/)(Kennedy, et al., 2001; Robinson, et al., 2005). The listing of murine MHC genes found at the IMGT web resource (http://www.ebi.ac.uk/imgt/) was the basis of the **Mouse_MHC_Allele** class and subclasses(Lefranc, 2001). The structure of the HLA Ontology is based on files provided via FTP by the IMGT/HLA database (http://www.ebi.ac.uk/imgt/hla/), as well as the HLA Dictionary for serological definitions (Schreuder, et al., 2005). Definitions of serological splits were provided by the website of the HLA Informatics Group at the Anthony Nolan Trust (http://www.anthonynolan.org.uk/HIG/lists/broad.html).

Applications

A website control for choosing alleles based upon the ontology was written using Java Server Pages in combination with AJAX and the database engine supplied by the Jena 2 API. A control for stand-alone applications for selecting multiple alleles was written in Visual Basic and query ontology entries in an InterSystems Caché database.

Results

The MHC Ontology can be accessed online under http://purl.org/stemnet/. It has also been submitted to the National Center for Biomedical Ontology's BioPortal (http://www.bioontology.org/bioportal.html). The major characteristics of the ontology are summarized in Table II. This first release of the MHC Ontology is given the version number 1.0. The HLA Ontology represents the HLA system as of the IMGT/HLA Database Release 2.20.0, 11 January 2008.

Table II – MHC Ontology facts sheet

Ontology Name	MHC Ontology	HLA Ontology
Namespace	http://purl.org/stemnet/MHC	http://purl.org/stemnet/HLA
Prefix	MHC	HLA
Scope	MHC alleles, genes and proteins in human, mouse, canine, and dog	HLA alleles
Format	OWL DL	OWL DL
Number of Classes	106	6649
Dependencies	Dublin core [a], HLA Ontology	Dublin core [a]
Data Sources	IPD [b], IMGT [c]	IMGT/HLA [d], HLA Dictionary [e], Anthony Nolan Trust [f]
Relations	*rdfs:subClassOf, encoded_in, encodes, has_part, part_of, variant_of, has_variant, from_species*	
Annotations	***rdfs:label, dc:creator, dc:date, dc:publisher, dc:source, dc:subject, dc:title, definition, synonym, reference***	
Additional Files		HLA.xml, HLA_twodigit.xml, HLA_twodigit.owl [g]

a. Dublin core: http://protege.stanford.edu/plugins/owl/dc/protege-dc.owl. **b**. Immuno Polymorphism Database(Robinson, et al., 2005). **c**. The international ImMunoGeneTics database(Lefranc, 2001). **d**. Official source of HLA nomenclature and sequences(Marsh, 2003). **e**. Source of serological associations(Schreuder, et al., 2005). **f**. Definitions of serological splits were provided by the website of the HLA Informatics Group at the Anthony Nolan Trust (http://www.anthonynolan.org.uk/HIG/lists/broad.html). **g**. These files can be accessed at http://purl.org/stemnet/

The MHC ontology consists of 106 classes and 7 relations. The MHC Ontology's namespace is http://purl.org/stemnet/MHC. Additionally, the imported HLA Ontology contains 6649 classes. Its namespace is http://purl.org/stemnet/HLA, and has been given the prefix "HLA" when imported into the MHC Ontology. The relations are summarized in Table III and include *from_species*, *encoded_in*, *has_part*, *has_variant* along with their respective inverse relations.

Table III – Relations of the MHC Ontology

Relation	Domain	Range	Properties	Example
encoded_in	**Polypeptide, Protein**	**Nuceleotide_Sequence**	Inverse of *encodes*	**MHC_Chain** *encoded_in* **MHC_Allele**
encodes	**Nucleotide_Sequence**	**Polypeptide, Protein**	Inverse of *encoded_in*	**MHC_Allele** *encodes* **MHC_Chain**
has_part	**owl:thing**	**owl:thing**	Inverse of *part_of*	**MHC_Protein** *has_part* **MHC_Chain**
part_of	**owl:thing**	**owl:thing**	Inverse of *has_part*	**MHC_ClassII_Beta** *part_of* **MHC_ClassII_Protein**
variant_of	**Allele**	**Gene**	Inverse of *has_variant*	**MHC_Allele** *variant_of* **MHC_Gene**
has_variant	**Gene**	**Allele**	Inverse of *variant_of*	**MHC_Gene** *has_variant* **MHC_Allele**
from_species	**Protein, Nucleotide_Sequence, Polypeptide**	**Organism**	Functional	**HLA_Class_I_Allele** *from_species* **Human**

To ensure compatibility with other established ontologies, and to prevent redundancy, many of the classes of the MHC Ontology are linked to external ontology entries via ***reference*** annotation statements. MHC Ontology classes and their corresponding classes in external ontologies are listed in Table IV. Currently, the only reference to an immunogenetic resource is **MHC Molecule** in the ontology of the Immune Epitope Database which is reference by the class **MHC_Protein** in the MHC Ontology (Sathiamurthy, et al., 2005). Many of the MHC Ontology entries correspond to terms in the Sequence Ontology (SO) (Eilbeck, et al., 2005). Further reference terms were found in the NCI Thesaurus, as well as in the ontology of Chemical Entities of Biological Interest (ChEBI) (http://www.ebi.ac.uk/**chebi**/) (Degtyarenko, et al., 2008). The organism classes of the MHC Ontology are linked to entries in NCBI's Taxonomy database (www.**ncbi**.nlm.nih.gov/**Taxonomy**/) (Wheeler, et al., 2000).

Table IV – References to external ontologies

MHC Ontology Class	External Ontology Class(es)
MHC Protein	IEDB: **MHC Molecule**
Allele	NCI:C16277 **Allele**, SO:0001023 **allele**
Gene	SO:0000704 **gene**
Pseudogene	SO:0000336 **pseudogene**
Protein	CHEBI:36080 **proteins**
Polypeptide	SO:0000104 **polypeptide**
Chain	CHEBI:16541 **protein polypeptide chains**, SO:0001063 **immature_peptide_region**
Jawed_Vertebrates	TaxonomyID:7776 **Gnathostomata**
Human	TaxonomyID:9606 **Homo sapiens**
Dog	TaxonomyID:9615 **Canis lupus familiaris**
Mouse	TaxonomyID:10090 **Mus musculus**
Organism	NCI:C14250 **Organism**

IEDB = Immune Epitope Database. NCI = The National Cancer Institute Thesaurus; SO = Sequence Ontology; CHEBI = Chemical Entities of Biological Interest; TaxonomyID = These are IDs given to entries in the NCBI Taxonomy database.

Hierarchical structure of HLA alleles

The hierarchical structure of HLA alleles, as described in the introduction and Table I, was implemented in the HLA Ontology. Alleles belonging to an HLA locus were divided into serological groups and then subdivided into serological splits. In parallel, the other typical categorization of HLA was implemented: a categorization based upon the number of digits of the allele name known to the observer. Two-digit alleles are siblings of serological groups. Two-digit alleles as well as serological groups are then subdivided into four-digit, six-digit, and finally eight-digit allelic classes. The following exceptions to this rule were made: 1) Alleles whose names begin with A*92 were placed in the A*02 group. 2) Alleles whose names begin with B*95 were placed within the B*15 group. 3) The two-digit group levels for MIC, TAP, and DP were excluded.

Representation of the MHC and HLA Ontology

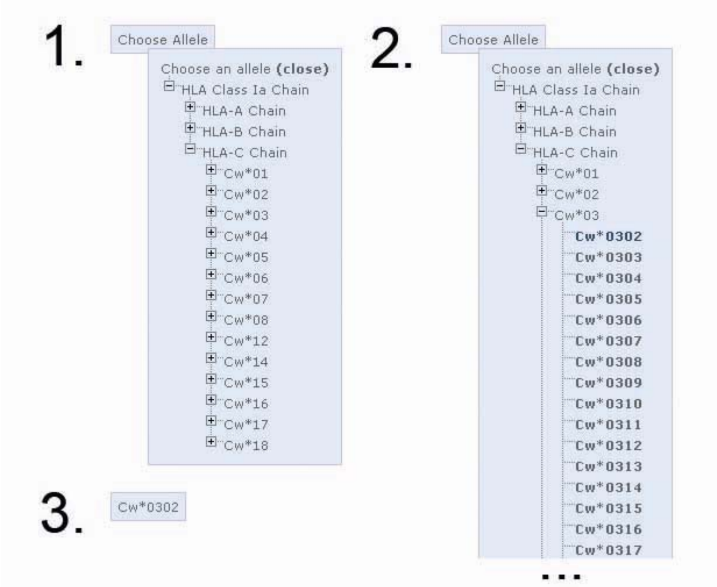
The MHC Ontology is represented in two files: MHC.owl and HLA.owl.
An additional file HLA_twodigit.owl is provided that represents the HLA Ontology without the serological group classes. Because serological typing is becoming a legacy technology the serological groups in the HLA Ontology are necessary in some applications, but obsolete in other. For example, serological groups are often mentioned in the literature and therefore these texts must be annotated using serological groups. Sequencing software, on the other hand, has no use for serological groups, and filtering them out dynamically could hinder performance.

Our experience has shown that additional formats can be very convenient for representing the ontological data. Thus, additional files, HLA.xml, HLA_twodigit.owl, and HLA_twodigit.xml can be found under http://purl.org/stemnet/. These files provide non-OWL XML representations of the HLA Ontology, which can be practical for bioinformatians and programmers which are not interested in carrying the overhead of OWL-parsing libraries. XML parsing libraries on the other hand are ubiquitous. Although OWL is a form of XML, the representations in HLA.xml and HLA_twodigit.xml are much simpler and more intuitive than the OWL equivalents and should be used for purposes for which the expressiveness of OWL is not needed.

User interfaces

Once the relations between HLA concepts had been formally defined in the computer-readable OWL format, creating user interfaces for HLA-centered programs become much easier. We demonstrated this by creating a new webpage dialog window which displays the content of an ontology so that the user may choose one entity (see Figure 1).

Figure 1 – A web popup control for choosing ontology elements

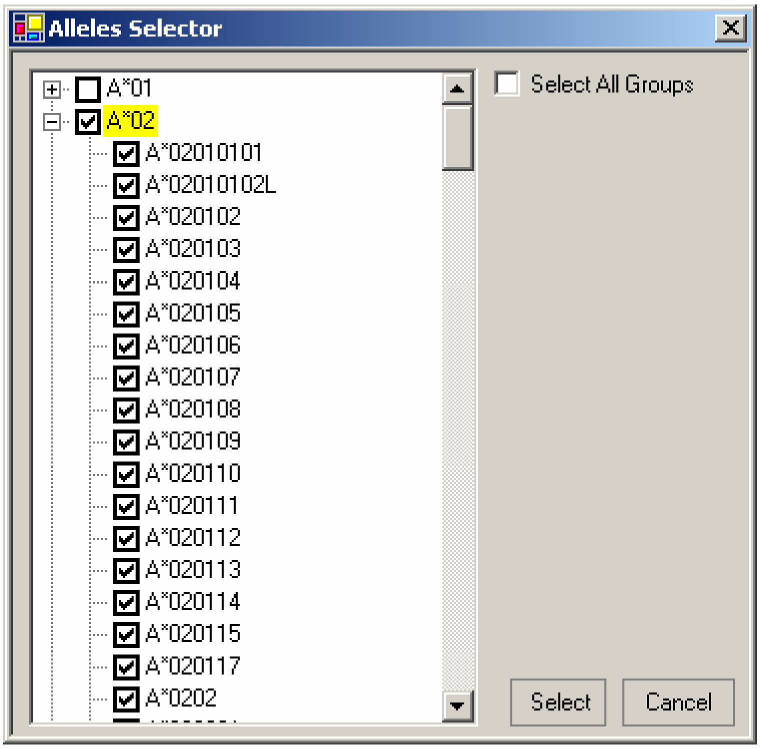


The allele chooser receives a top class as a parameter. In this case, Class I chains are of interest and so the top element is **HLA_Class_Ia_Chain**. In step 1, the user clicks on the “Choose Allele” button to reveal a tree structure in which subclasses can be opened. In step 2, the user has reached the leaf level and may select the desired chain. Step 3 shows that the user’s selection has been assigned, and the program can use this input for whatever purpose.

This dialog solved the problem of choosing a specific HLA allele or protein from a list of thousands of entries. The hierarchical nature of ontologies allows them to be represented as a tree structure. In this way, the user is not overwhelmed by a long list of allele or protein names, but simply opens the categories of interest, revealing only the relevant subset. This also has performance benefits with respect to the time it takes to load a webpage. Since the dialog is based upon AJAX, only the relevant information is loaded with each click by communicating with the server in the background. This saves the significant lag time required to load the entire ontology into the browser, especially when dealing with large ontologies. The dialog is not only functional for the HLA Ontology, but can load any ontology which has been represented in OWL. Other formats could be incorporated without requiring major modifications. This dialog can be seen in use on the www.peptidecheck.de webpage, under the Module Explorer page, and also at www.histocheck.de when choosing proteins for comparison.

In some applications, the user wants to perform an action on a group of alleles. This is particularly true when working at low resolution – or on the “two-digit” group level. For this purpose we have developed a Visual Basic-based dialog which uses the HLA ontology to allow the user to conveniently select multiple alleles (see figure).

Figure 2 – A visual basic control for selecting multiple classes in an ontology



□ A*01
☑ A*02
☑ A*02010101
☑ A*02010102L
☑ A*020102
☑ A*020103
☑ A*020104
☑ A*020105
☑ A*020106
☑ A*020107
☑ A*020108
☑ A*020109
☑ A*020110
☑ A*020111
☑ A*020112
☑ A*020113
☑ A*020114
☑ A*020115
☑ A*020117
☑ A*0202

□ Select All Groups

The advantage of this ontology-driven dialog becomes clear when one wants to perform an action involving all A*02 alleles. Because the A*92 alleles are also part of the A*02 group as defined in the ontology, they will also be included when choosing the whole of A*02.

Database schema

The PeptideCheck and HistoCheck websites are now driven by the MHC Ontology. Currently each allele entry in the database is tagged with its appropriate entry in the ontology. This allows for the use of allele chooser mentioned above. The *encoded_in* relation is an important link in the database between the entries of HLA chains and alleles. Because the HistoCheck website depicts entire HLA Proteins, consisting of alpha and beta chains for class II, the relations *has_part* for subclasses of **MHC_Class_II_Protein** has proven valuable.

Discussion

One of the first challenges when examining the semantics of MHC is the confusion between MHC as a region of the genome, and MHC as a protein on the surface of cells. It is common to hear or read something like "the peptide is bound by MHC class I". Figures depicting the proteins involved in peptide presentation often contain the labels, *T cell receptor*, *peptide* and "*MHC*". However, MHC stands for major histocompatibility complex, and as such, it would me more accurate to reserve the term MHC to denote a gene complex, and not a protein. The MHC is a gene region containing genes which encode peptide presenting proteins, as well as other kinds of proteins, such as those involved in the proteasome. This distinction must be made consistently, and for this reason we have included classes such as **Human_MHC_Class_I_Region_Allele** in the MHC Ontology. The class I region is a portion of the MHC which contains genes such has HLA-A, B, C, etc, but also MIC. Although MIC genes are not HLA Class I genes (MIC is considered "class I-similar"), they are encoded in the Human MHC Class I Region. The term HLA in this ontology refers to genes which encode proteins which present peptides on the surface of cells. The analog in mouse is of course, H2, and DLA in dogs. The term chosen in the ontology for these alleles which is organism independent is **MHC_Allele_Encoding_Peptide_Presenting_Protein**.

Furthermore, the scientific community rarely makes a distinction in nomenclature between gene names and the names of the proteins encoded by those genes. The HLA nomenclature refers only to alleles, and as such there is no distinct terminology for HLA proteins or chains. For the ontology, we have appended the word "Chain" to each appropriate class name (e.g. **B_4402_Chain**). The relevant ***rdfs:label*** however does not contain the word chain (e.g. B*4402), and is, outside of the hierarchical context, indistinguishable from the allele label.

The concept of expression in biology is another point of difficulty. Gene expression is defined by the NCI Thesaurus as "Typically involves transcription of genetically encoded information into an intermediary message (messenger RNA) and subsequent translation into a functional protein." In this case, it is not entirely clear whether null alleles are expressed. They could very likely be transcribed and translated into functionless peptide chains which would soon be digested. However, we choose to consider such alleles as non-expressed and have classified them as **Null_Allele**, a direct child of **Allele**. Accordingly, there are no members of the class **Human_MHC_Chain**, encoded by **Null_Allele**s. The HLA nomenclature accounts for five forms of alternatively expressed alleles. Of the classes included in **Alternatively_Expressed_Allele**, we have chosen to only create the correlatives **HLA_Cytoplasm_Chain**, **HLA_Low_Chain**, **HLA_Secreted_Chain** under the class **Alternatively_Expressed_Chain** (see Table V). Because there is no conclusive experimental evidence describing chain products of alleles included in **HLA_Aberrant_Allele** and **HLA_Questionable_Allele**, we have excluded equivalent classes under **Chain**.

Table V – Alternatively Expressed Alleles and Chains

Class	Subclass	Definition
Alternatively_Expressed_Allele		Alternatively expressed HLA alleles encode proteins, whose protein sequence are not drastically affected by the mutation, but contain mutations which lead to sub-normal cell surface expression. HLA alleles of this nature are named with suffixes 'L', 'S', 'C', 'A' or 'Q'
	HLA_Aberrant_Allele	An allele whose expression is aberrant meaning that there is some doubt as to whether a protein is expressed.
	HLA_Cytoplasm_Allele	An allele which encodes a protein which collects in the cytoplasm after translation and is not expressed on the surface of the cell.
	HLA_Low_Allele	An allele which encodes a protein which is represented only at low levels on the surface of the cell.
	HLA_Questionable_Allele	Allele whose expression is questionable given that the mutation seen in the allele has previously been shown to effect normal expression levels
	HLA_Secreted_Allele	An allele which encodes a protein which is soluble and secreted from the cell, but not present on the cell surface.
Alternatively_Expressed_Chain		Chain encoded by an alternatively expressed allele
	HLA_Cytoplasm_Chain	A chain which collects in the cytoplasm after translation and is not expressed on the surface of the cell.
	HLA_Cytoplasm_Chain	A chain which is represented only at low levels on the surface of the cell.
	HLA_Secreted_Chain	A chain which is soluble and secreted from the cell, but not present on the cell surface.

The context of the MHC Ontology within the community of Biomedical Ontologies must be attentively considered. This underscores the importance of Table IV. Further immunogenetically relevant ontologies include the IEDB(Sathiamurthy, et al., 2005) ontology and the IMGT Ontology(Giudicelli, et al., 2005). After inspection of these ontologies, it is clear that overlap with the MHC Ontology is minimal. While these Ontologies complement each other effectively at the moment, as these ontologies grow, vigilance is required to avoid redundancy and ensure compatibility.

These results show that ontologies improve the organization of complex concepts in immunogenetics. In the long term, development of an international standard would lead to the homogeneous database structures in centers and labs across the world. Utilizing such ontologies has a high potential for ensuring efficient networking and collaboration. A spectrum of immunoinformatic tools for the MHC system has been established, and will continue to grow. Applications include typing software, bone marrow registry analysis (Muller, 2002), histocompatibility prediction (Elsner, et al., 2004), T cell and B cell epitope prediction (Buus, et al., 2003; Duquesnoy and Askar, 2007), and minor histocompatibility antigen prediction algorithms (Halling-Brown, et al., 2006; Schuler, et al., 2005). When serving as an infrastructure upon which further immunoinformatic tools can be built, the MHC Ontology will allow such tools would be easily integrated for use in research institutes and clinical laboratories.

Acknowlegements

I would like to thank the members of the Data Interoperability Steering Committee Meeting of the Division of Allergy, Immunology, and Transplantation at NIH, in particular, the members of the HLA subcommittee, Bjoern Peters, Steven Marsh, and Richard Scheuermann.

REFERENCES

Buus, S., Lauemoller, S.L., Worning, P., Kesmir, C., Frimurer, T., Corbet, S., Fomsgaard, A., Hilden, J., Holm, A. and Brunak, S. (2003) Sensitive quantitative predictions of peptide-MHC binding by a 'Query by Committee' artificial neural network approach, *Tissue Antigens*, **62**, 378-384.

Carroll, J.J., Dickinson, I., Dollin, C., Reynolds, D., Seaborne, A. and Wilkinson, K. (2004) Jena: implementing the semantic web recommendations, *Proceedings of the 13th international World Wide Web conference on Alternate track papers & posters*, 74 - 83.

Degtyarenko, K., de Matos, P., Ennis, M., Hastings, J., Zbinden, M., McNaught, A., Alcantara, R., Darsow, M., Guedj, M. and Ashburner, M. (2008) ChEBI: a database and ontology for chemical entities of biological interest, *Nucleic Acids Res*, **36**, D344-350.

Duquesnoy, R.J. and Askar, M. (2007) HLAMatchmaker: a molecularly based algorithm for histocompatibility determination. V. Eplet matching for HLA-DR, HLA-DQ, and HLA-DP, *Hum Immunol*, **68**, 12-25.

Eilbeck, K., Lewis, S.E., Mungall, C.J., Yandell, M., Stein, L., Durbin, R. and Ashburner, M. (2005) The Sequence Ontology: a tool for the unification of genome annotations, *Genome Biol*, **6**, R44.

Elsner, H.A., DeLuca, D., Strub, J. and Blasczyk, R. (2004) HistoCheck: rating of HLA class I and II mismatches by an internet-based software tool, *Bone Marrow Transplant*, **33**, 165-169.

Giudicelli, V., Chaume, D., Jabado-Michaloud, J. and Lefranc, M.P. (2005) Immunogenetics Sequence Annotation: the Strategy of IMGT based on IMGT-ONTOLOGY, *Stud Health Technol Inform*, **116**, 3-8.

Hahn, U., Wermter, J., Blasczyk, R. and Horn, P.A. (2007) Text mining: powering the database revolution, *Nature*, **448**, 130.

Halling-Brown, M., Quartey-Papafio, R., Travers, P.J. and Moss, D.S. (2006) SiPep: a system for the prediction of tissue-specific minor histocompatibility antigens, *Int J Immunogenet*, **33**, 289-295.

Kennedy, L.J., Angles, J.M., Barnes, A., Carter, S.D., Francino, O., Gerlach, J.A., Happ, G.M., Ollier, W.E., Thomson, W. and Wagner, J.L. (2001) Nomenclature for factors of the dog major histocompatibility system (DLA), 2000: Second report of the ISAG DLA Nomenclature Committee, *Tissue Antigens*, **58**, 55-70.

Lefranc, M.P. (2001) IMGT, the international ImMunoGeneTics database, *Nucleic Acids Res*, **29**, 207-209.

Little, A.M. (2007) An overview of HLA typing for hematopoietic stem cell transplantation, *Methods Mol Med*, **134**, 35-49.

Marsh, S.G. (2003) HLA nomenclature and the IMGT/HLA sequence database, *Novartis Found Symp*, **254**, 165-173; discussion 173-166, 216-122, 250-162.

Marsh, S.G.E., Albert, E.D., Bodmer, W.F., Bontrop, R.E., Dupont, B., Erlich, H.A., Geraghty, D.E., Hansen, J.A., Mach, B., Mayr, W.R., Parham, P., Petersdorf, E.W., Sasazuki, T., Schreuder, G.M.T., Strominger, J.L., Svejgaard, A. and Terasaki, P.I. (2002) Nomenclature for factors of the HLA system, 2002, *Human Immunology*, **63**, 1213-1268.

Muller, C.R. (2002) Computer applications in the search for unrelated stem cell donors, *Transpl Immunol*, **10**, 227-240.

Robinson, J., Waller, M.J., Stoehr, P. and Marsh, S.G. (2005) IPD--the Immuno Polymorphism Database, *Nucleic Acids Res*, **33**, D523-526.

Sathiamurthy, M., Peters, B., Bui, H.H., Sidney, J., Mokili, J., Wilson, S.S., Fleri, W., McGuinness, D.L., Bourne, P.E. and Sette, A. (2005) An ontology for immune epitopes: application to the design of a broad scope database of immune reactivities, *Immunome Res*, **1**, 2.

Schreuder, G.M., Hurley, C.K., Marsh, S.G., Lau, M., Fernandez-Vina, M., Noreen, H.J., Setterholm, M. and Maiers, M. (2005) The HLA Dictionary 2004: a summary of HLA-A, -B, -C, -DRB1/3/4/5 and -DQB1 alleles and their association with serologically defined HLA-A, -B, -C, -DR and -DQ antigens, *Tissue Antigens*, **65**, 1-55.

Schuler, M.M., Donnes, P., Nastke, M.D., Kohlbacher, O., Rammensee, H.G. and Stevanovic, S. (2005) SNEP: SNP-derived epitope prediction program for minor H antigens, *Immunogenetics*, **57**, 816-820.

Smith, B., Ashburner, M., Rosse, C., Bard, J., Bug, W., Ceusters, W., Goldberg, L.J., Eilbeck, K., Ireland, A., Mungall, C.J., Leontis, N., Rocca-Serra, P., Ruttenberg, A., Sansone, S.A., Scheuermann, R.H., Shah, N., Whetzel, P.L. and Lewis, S. (2007) The OBO Foundry: coordinated evolution of ontologies to support biomedical data integration, *Nat Biotechnol*, **25**, 1251-1255.

Wheeler, D.L., Chappey, C., Lash, A.E., Leipe, D.D., Madden, T.L., Schuler, G.D., Tatusova, T.A. and Rapp, B.A. (2000) Database resources of the National Center for Biotechnology Information, *Nucleic Acids Res*, **28**, 10-14.

Part IV. Appendix

A Data Warehousing

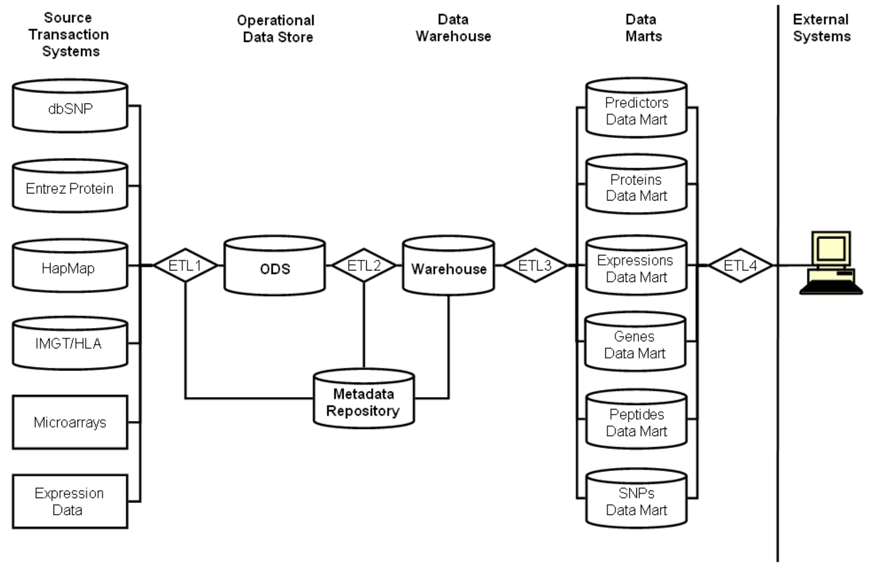
To instantiate PeptideCheck, a system of databases and algorithms had to be implemented with a large level of complexity. To manage such complex systems, computer scientists have developed the concept of data warehousing. A data warehouse is a network of data sources and repositories in combination with algorithms which transfer and restructure the data. Data warehousing as a concept is not a magic bullet solution complex systems, but more of a logical approach borne out of the necessity faced by any programmer dealing with a large amount of diverse data. A schema of the data warehousing involved in PeptideCheck is found in Figure 1.

Figure 1 depicts a list of data sources on the left side which are processed by the first extract transformation load (ETL1). The ETL1 is responsible for simply acquiring the data from these sources. For example, in the case of SNP data from the dbSNP the ETL1 is a program which connects to the dbSNP via the HTTP Internet protocol, and downloads the data to the PeptideCheck server and stores it as a series of XML files. These files represent the operational data store (ODS). The ODS is the primary temporary storage location for data coming into PeptideCheck. Another example of data in the ODS is the genotypic frequency data from HapMap in the form of ZIP archives.

The ETL2 must extract data from the ODS and store it in the PeptideCheck database as raw data. This portion of the database, containing raw, unprocessed data is known as the warehouse. Examples of ETL2 processing include XML parsing in the case of dbSNP and Entrez Protein data, or MSF (sequence alignment) parsing in the case of the IMGT/HLA database.

The ETL3 is responsible for reorganizing the data in the warehouse into a form of storage which is clean, intuitive, and efficient for querying. In PeptideCheck, the ETL3 for SNP processing is one of the most complicated pieces

Fig. 1: PeptideCheck as Data Warehouse



of programming. The reason for this is that the SNP data in the dbSNP is not organized around biological principles, but rather around the processes of data submission by scientists. The dbSNP entries (so-called rs entries) are specific to a particular gene, but can contain multiple types of polymorphism at multiple locations, as well as duplications and errors. For example, the ETL3 must check whether the reported reference amino acid can actually be found at the reported position in the protein sequence. Furthermore, we have chosen the form of SNP storage which is “SNP-site oriented”. In other words, an SNP is defined as a position in a particular protein sequence containing a mutation (protein sequence, and not nucleotide sequence, since we are only interested in functional peptides). Another very important task performed by the ETL3 is the application of antigen presentation prediction algorithms, and the storage of the resulting scores.

The final data representation in PeptideCheck consists of a series of highly organized and cleansed tables, or Data Marts. A key requirement of the storage design is that the data can be queried quickly. This is particularly challenging, since the PeptideCheck user may choose any combination of prediction algorithms, thresholds, SNP requirements, etc. The solution was to create a system of custom database indexes, which is a unique feature of the InterSystems Caché database.

The ETL4 provides the final data transfer to the user. Since PeptideCheck is a web-based system, the ETL4 consists of the internet browser used to login to the website, as well as the Java Server Pages (JSPs) and Caché Server Pages (CSPs), and finally their connection to the database for querying. Server pages are programs which can response to internet browser requests via HTTP and provide dynamic content to the user. In the case of PeptideCheck, the dynamic content is produced by interpreting the input data of the user (e.g. peptide ranking/filtering criteria), querying the database, and responding with HTML output.

An additional aspect of the data warehousing concept is the Metadata Repository. This is a monitoring system, which is responsible for logging the actions of all the ETLs. Ideally, this provides the ability to track changes to the system during updates, to investigate the cause of errors, or to restore the system to a previous state. PeptideCheck's Metadata repository is limited to a series of log files, provided by the Log4J Java API from the open-source Apache Software Foundation. These log files provide a way of tracing through the actions performed by the system. However, it must be noted that there is presently no automated or convenient way to filter this data, or to use it for restoring previous states of data.

B Database Tables

The following is a list of tables found in the PeptideCheck database. PeptideCheck uses an InterSystems Caché database, which supports object oriented database design. The tables are organized into packages, called schema.

Schema: HLA

Table: Classes

Name	Type	Relation	Description
Id	Integer		Primary key
Name	String		Java class name of the the associated prediction algorithm
CacheProjection	String		Name of the Caché class which inherits from the Predictor.Interface
BitIndexName	String		Name of associated bitmap index
Group_Id	Integer	$\infty \rightarrow 1$: Groups.Id	Link to the Groups tabe, which describes which kind of predictor this is (HLA binding, proteasome cleavage, etc)

Table: AllelesClasses

Name	Type	Relation	Description
Id	Integer	$\infty \rightarrow 1$: HLAMatrixPredictor	Primary key
Name	String		Allele name (The nomenclature can differ between algorithms)
ClassId	Integer	$\infty \rightarrow 1$: Classes.Id	Link to the Class table

Table: Groups

Name	Type	Relation	Description
Id	Integer	1 → ∞ : Classes.Id	Primary key
Name	String		Name of the group (e.g. HLA Peptide Binding Prediction; Proteasomal Pross. Predict., etc)

Schema: PeptideCheck

Table: SNPS

Name	Type	Relation	Description
Id	Integer		Primary key
Proteins_Id	Integer	$\infty \to 1$: Proteins.Id	Link to the Proteins table
Rs_rsId	Integer		Id of an dbSNP entry
Rs_validation_ byFrequency	Boolean		True when validated by frequency
Rs_Validation_ byHapMap	Boolean		True when validated by HapMap
Component_ Chromosome	String(2)		Chromosome number or letter
GeneAlias_Id	Integer	$\infty \to 1$: GeneAlias_Id	The Alias id from the GeneAliases table
FxnSet_protAcc	String		Protein accession number from NCBI
FxnSet_residue	String(1)		One-letter amino acid code
FxnSet_aaPosition	Integer		SNP position in the protein sequence
FxnSet_allele	String		Nucleotide
AAList_Id	Integer	$\infty \to 1$: AAList.Id	Link to the AAList table
Type	TinyInteger		1= Missense; 2=Stop; 3=NonStop; 4= Insertion; 5= Deletion
Spec	Integer	$\infty \to 1$: SnpsSpec.id	Link to the table which identifies whether this entry refers to the reference allele

Table: SnpsSpec

Name	Type	Relation	Description
Id	Integer	$1 \rightarrow \infty$: Snps.Id	Primary key
Name	String		Name describing SNP entry type (reference, mutation, etc)

Table: Proteins

Name	Type	Relation	Description
Id	Integer	$\infty \rightarrow 1$: Snps.Id	Primary key
GBSeq_locus	String		The protein accession number from NCBI
GBSeq_sequence	String		The protein sequence

Table: GeneAliases

Name	Type	Relation	Description
Id	Integer		Primary key
Alias_Id	Integer	$\infty \rightarrow 1$: Snps.Id	This Id is associated with multiple gene symbols which refer to the same gene
External_Id	String		The numerical Id from Entrez Gene

Table: AAList

Name	Type	Relation	Description
Id	Integer	$\infty \rightarrow 1$: Snps.Id; $1 \rightarrow \infty$: PeptideDefinitions.Id	Primary key
PFValue	Float		HapMap frequencies converted to "peptide frequency" value*
Allele	String		The protein sequence
IsValidated	Boolean		True if validated by HapMap or by frequency as given in SNPs table
Residue	String		Amino acid
Type	Integer	$\infty \rightarrow 1$: AAListTypes.Id	Which type of SNP (Missence, Stop, Non-Stop, etc)
SNPSite_Id	Integer	$\infty \rightarrow 1$: SNPSite.Id	Link to the SNPSite table
Proteins_Id	Integer	$\infty \rightarrow 1$: Proteins.Id	Link to the Proteins table

* Note that in Section 9, this value is refered to as PP

Table: AAListTypes

Name	Type	Relation	Description
Id	Integer	$1 \rightarrow \infty$: AALists.Id	Primary key
Name	String		Name of this type of SNP

Table: SNPSite

Name	Type	Relation	Description
Id	Integer	$1 \rightarrow \infty$: AAList.Id	Primary key
Pos	Integer		The position in the protein referenced in the AAList
GeneAlias_Id	Integer	$\infty \rightarrow 1$: GeneAliases.Id	Link to the gene alias

Table: PeptideCandidates

Name	Type	Relation	Description
Id	Integer	$1 \rightarrow \infty$: PeptideDefinitions.Id	Primary key
Peptide	String		The peptide sequence
GeneAliasId	Integer	$\infty \rightarrow 1$: GeneAliases.Id	Link to the gene alias

Schema: Predictors

Interface: AllelePredictors

Name	Type	Relation	Description
AlleleId	Integer	$1 \rightarrow \infty$: AllelesClasses.Id	Link to the AlellesClasses table
ScoreStarts	Integer		The starting score used for creating the ranged index
ScoreEnds	Integer		The ending score fro the ranged index
PeptideBitMap	BitMap Index		Index containing the Ids of the PeptideCandidates

Interface: CleavagePredictor

Name	Type	Relation	Description
ScoreStarts	Integer		The starting score used for creating the ranged index
ScoreEnds	Integer		The ending score fro the ranged index
PeptideBitMap	BitMap Index		Index containing the Ids of the PeptideCandidates

Schema: Expression

Table: ExpressionData

Name	Type	Relation	Description
Id	Integer		Primary key
CellType	Integer	$\infty \rightarrow 1$: CellType.Id	Link to the CellType entry
GeneAlias_Id	Integer	$\infty \rightarrow 1$: GeneAliases.Id	Link to the gene

Table: CellType

Name	Type	Relation	Description
Id	Integer	$1 \rightarrow \infty$: ExpressionData.Id, Affymetrix.Id, AffyResults.Id	Primary key
Name	String		Name of the cell type (e.g. CML)

Table: AffyMetrix

Name	Type	Relation	Description
Id	Integer		Primary key
Affy_Id	String	$1 \rightarrow \infty$: AffyResults.Id	This is the ID provided by the affymetrix result file
CellType	Integer	$\infty \rightarrow 1$: CellType.Id	Link to the CellType table (e.g. CML)
GeneAlias_Id	Integer	$\infty \rightarrow 1$: GeneAliases.Id	Link to gene
IsDataUnLogical	Boolean		Flag, true when this Affymetrix result could be linked to a gene
Information	String		The raw text provided for the Affymetrix result
GeneSymbol	String		The gene name parsed from the result file

Table: AffyResults

Name	Type	Relation	Description
Id	Integer		Primary key
Affy_Id	Integer	$\infty \rightarrow 1$: AffyMetrix.Id	Link to the AffyMetrix table
CellType	Integer	$\infty \rightarrow 1$: CellType.Id	Link to the CellType table (e.g. CML)
Signal	Float		Signal strength from the probe detection
Detected	Boolean		True when a signal was provided

C Prediction Algorithms and Inheritance

When developing PeptideCheck, it was expected that a number of peptide binding and processing prediction algorithms would be incorporated. It was also expected, that these algorithms could become outdated, as prediction strategies improved. However, because such algorithms always serve the same function within PeptideCheck, it made sense to build the system such that new algorithms could be inserted as modules (or plug-ins), with minimal additional programming. The concept in object oriented programming that allows us to do this is inheritance. Inheritance is supported both by Java, and by the InterSystems Caché database.

There are two categories of prediction algorithms relevant to PeptideCheck: HLA allele associated, and non-HLA allele associated (e.g. proteasomal processing prediction). Both types of prediction require a peptide sequence as input, and produce a prediction score as output. The difference for the HLA binding prediction is that each score must also be associated with a specific HLA allele. Proteasomal processing, on the other hand, is HLA independent.

Java

In java, an interface was created called PeptidePredictor. This defines the core functionally of a prediction algorithm.

Java Interface: PeptidePredictor

Return Value	Method Name
PParams	getParameters()
double	getScore(java.lang.String seq)
void	init(PParams _params, DDatabase db)
boolean	isPositive(double score)
void	setParameters(PParams params)

The PParams object is a structure that contains prediction settings (e.g.

threshold). Because most algorithms must be initialized by either connecting to a database, or loading some kind of initialization data from a file, the init method has also defined in this interface. The isPositive method returns a boolean value as a prediction result (e.g. is a binder, is not a binder). Typically, the value is found by comparing the threshold in the PParams object to the result of the getScore method.

In line with the goal of minimizing programming when adding new algorithms, the next step was the implementation of an abstract class. The abstract class allows for the implementation of methods which are expected to be the same for all extending classes, but permits that some methods are left unimplemented.

Java Abstract Class: AbstractPredictor implements PeptidePredictor

Return Value	Method Name
PParams	getParameters()
boolean	isPositive(double score)
void	setParameters(PParams params)
void	setThreshold(double t)

It should be noted that the methods init and getScore from the PeptidePredictor interface are not implemented in AbstractPredictor. These methods must remain abstract because they are different for each prediction algorithm. The methods getParameters, isPositive, and setParameters are implemented because their behavior is common to all prediction algorithms. In the event that a particular algorithm requires different behavior, the method can of course be overridden in the final class.

The AbstractPredictor class can be extended to a specific proteasomal cleavage algorithm. For HLA peptide binding prediction algorithms, however, special behavior is required because these algorithms are associated with specific HLA alleles. Therefore, an additional abstract classes was created which is specific to peptide binding algorithms: AbstractPBPredictor.

Java Abstract Class: AbstractPBPredictor implements AbstractPredictor

Return Value	Method Name
int	compareTo(java.lang.Object o)
Collection <PParams>	getPossiblePredictions(DDatabase _db)

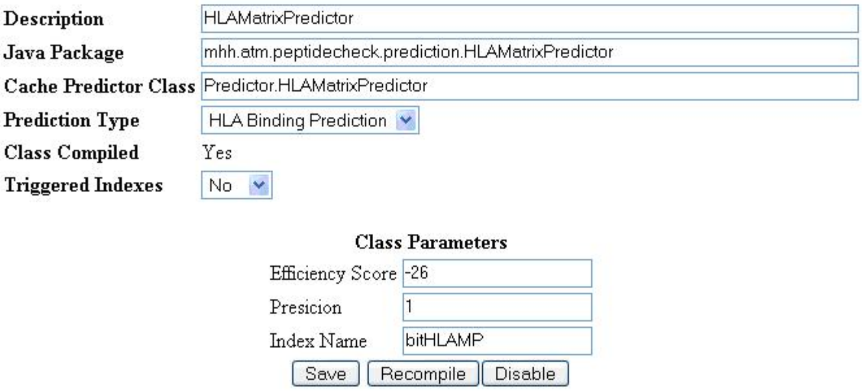
Note that AbstractPBPredictor implements AbstractPredictor, because the basic functions provided by AbstractPredictor are useful for algorithms involving HLA alleles as well. The compareTo method was added, which allows the algorithms to be sorted alphabetically by allele name. This can be convenient for displaying the prediction results in the correct format. The other addition method is the getPossiblePredictions method, which returns a collection of PParams. This method is useful because PeptideCheck must often make predictions for every available HLA allele.

Caché

The database must also be able to cope with new prediction algorithms. Fortunately, the Caché database provides for object orientation. To add a new java-based prediction algorithm to the database, an entry in the HLA.Classes table must be made (See Appendix B). To expedite this process, a web interface was created allowing the administrator to easily enter the appropriate information (Figure 2).

The class parameters are important values for generating the indexes on the prediction scores. When the information is saved, the a new Caché class is generated by the Class Generator (provided by Caché), which implements either the Predictors.AllelePredictor or Predictors.CleavagePredictor interface as defined in Appendix B. The prediction classes can then be managed by the web interface depicted in Figure 3.

Fig. 2: Adding a new prediction algorithm

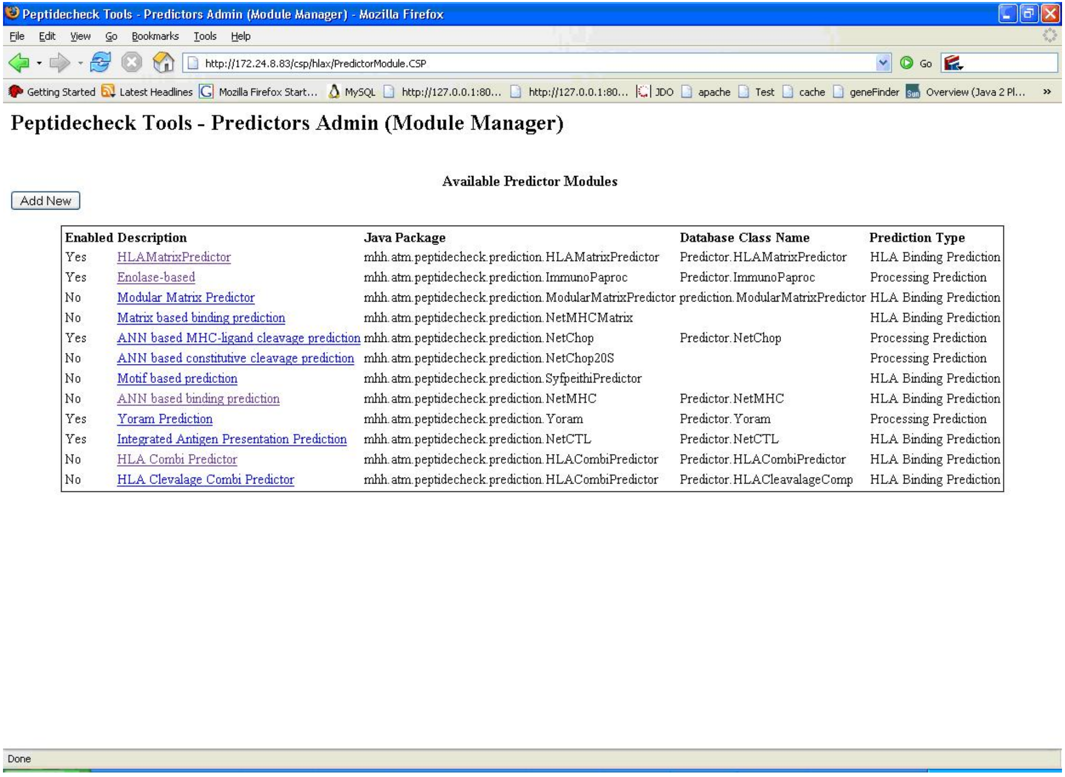


Description	HLAMatrixPredictor
Java Package	mhh.atm.peptidecheck.prediction.HLAMatrixPredictor
Cache Predictor Class	Predictor.HLAMatrixPredictor
Prediction Type	HLA Binding Prediction
Class Compiled	Yes
Triggered Indexes	No

Class Parameters

Efficiency Score	-26
Presicion	1
Index Name	bitHLAMP

Fig. 3: Administration of prediction algorithms



Peptidecheck Tools - Predictors Admin (Module Manager)

Available Predictor Modules

Enabled	Description	Java Package	Database Class Name	Prediction Type
Yes	HLAMatrixPredictor	mhh.atm.peptidecheck.prediction.HLAMatrixPredictor	Predictor.HLAMatrixPredictor	HLA Binding Prediction
Yes	Enolase-based	mhh.atm.peptidecheck.prediction.ImmunoPaproc	Predictor.ImmunoPaproc	Processing Prediction
No	Modular Matrix Predictor	mhh.atm.peptidecheck.prediction.ModularMatrixPredictor	prediction.ModularMatrixPredictor	HLA Binding Prediction
No	Matrix based binding prediction	mhh.atm.peptidecheck.prediction.NetMHCMatrix		HLA Binding Prediction
Yes	ANN based MHC-ligand cleavage prediction	mhh.atm.peptidecheck.prediction.NetChop	Predictor.NetChop	Processing Prediction
No	ANN based constitutive cleavage prediction	mhh.atm.peptidecheck.prediction.NetChop20S		Processing Prediction
No	Motif based prediction	mhh.atm.peptidecheck.prediction.SyfpeithiPredictor		HLA Binding Prediction
No	ANN based binding prediction	mhh.atm.peptidecheck.prediction.NetMHC	Predictor.NetMHC	HLA Binding Prediction
Yes	Yoram Prediction	mhh.atm.peptidecheck.prediction.Yoram	Predictor.Yoram	Processing Prediction
Yes	Integrated Antigen Presentation Prediction	mhh.atm.peptidecheck.prediction.NetCTL	Predictor.NetCTL	HLA Binding Prediction
No	HLA Combi Predictor	mhh.atm.peptidecheck.prediction.HLACombiPredictor	Predictor.HLACombiPredictor	HLA Binding Prediction
No	HLA Clevalage Combi Predictor	mhh.atm.peptidecheck.prediction.HLACombiPredictor	Predictor.HLACleavalageComp	HLA Binding Prediction

D Declaration

I hereby swear that I have composed this dissertation independently, and that all aid and support from any recruited institutions have been identified here within. Furthermore, I ensure that the dissertation was not used to obtain any other university degrees.

Hannover, May 22st, 2008

David S. DeLuca

E Curriculum Vitae

Name:		David S. DeLuca	
Birthday		July 2nd, 1980	
Birth Place:		Pittsfield, MA, USA	
Primary Education	1985	Egremont Elementary School	
	1986 - 1990	Stearns Elementary School	
	1990 - 1994	Herberg Middle School	
	1994 - 1998	Pittsfield High School	
Secondary Education	1998 - 2002	Carnegie Mellon University	B.Sc.
	2002 - 2005	Universität Hannover	M.Sc.
Employment	2002 - 2008	Institute für Transfusionsmedizin, Medizinische Hochschule Hannover	

F Publications of Author

1. DeLuca, D.S., Beisswanger, E., Hahn, U. and Blasczyk, R. (2007) An automatically curated MHC ontology for intelligent immunogenetic information retrieval, J Immunol, 178, LB9-.

2. DeLuca, D.S. and Blasczyk, R. (2007) HistoCheck: Evaluating Structural and Functional MHC Similarities. In Flower, D.R. (ed), Immunoinformatics. Humana Press, 395-405.

3. DeLuca, D.S. and Blasczyk, R. (2007) The immunoinformatics of cancer immunotherapy, Tissue Antigens, 70, 265-271.

4. DeLuca, D.S and Blasczyk, R. (2007) Implementing the Modular MHC Model for Predicting Peptide Binding. In Flower, D.R. (ed), Immunoinformatics. Humana Press, 261-271.

5. DeLuca, D.S., Khattab, B. and Blasczyk, R. (2007) A modular concept of HLA for comprehensive peptide binding prediction, Immunogenetics, 59, 25-35.

6. Eiz-Vesper, B., Deluca, D.S., Blasczyk, R. and Horn, P.A. (2007) The nature of recombination in HLA-B*4207, Tissue Antigens, 70, 164-168.

7. Bade-Doeding, C., DeLuca, D.S., Seltsam, A., Blasczyk, R. and Eiz-Vesper, B. (2007) Amino acid 95 causes strong alteration of peptide position Pomega in HLA-B*41 variants, Immunogenetics, 59, 253-259.

8. Seltsam, A., Strigens, S., Levene, C., Yahalom, V., Moulds, M., Moulds, J.J., Hustinx, H., Weisbach, V., Figueroa, D., Bade-Doeding, C., DeLuca, D.S. and Blasczyk, R. (2007) The molecular diversity of Sema7A, the semaphorin that carries the JMH blood group antigens, Transfusion, 47, 133-146.

9. Alkharsah, K.R., Dedicoat, M., DeLuca, D.S., Schulz, T.F. and Blasczyk, R. (2006) Identification of two new HLA-A variants, HLA-A*2911 and

HLA-A*6827, Tissue Antigens, 67, 170-172.

10. Horn, P.A., DeLuca, D.S. and Blasczyk, R. (2005) An amino acid contributing to pockets A, B, and D of the peptide-binding groove is altered in A*0315, Tissue Antigens, 66, 703-704.

11. Horn, P.A., DeLuca, D.S., Jindra, P. and Blasczyk, R. (2005) The replacement mutation in HLA-DRB1*1211 affects a likely keystone position, Hum Immunol, 66, 1254-1257.

12. Horn, P.A., DeLuca, D.S., Mueller, K. and Blasczyk, R. (2005) Peptide-binding characteristics of the novel allele DRB1*0112 are probably identical to DRB1*0101, Tissue Antigens, 65, 505-506.

13. Elsner, H.A., DeLuca, D., Strub, J. and Blasczyk, R. (2004) HistoCheck: rating of HLA class I and II mismatches by an internet-based software tool, Bone Marrow Transplant, 33, 165-169.