

# *NetCTLpan*: pan-specific MHC class I pathway epitope predictions

Thomas Stranzl · Mette Voldby Larsen ·  
Claus Lundegaard · Morten Nielsen

Received: 31 October 2009 / Accepted: 16 March 2010 / Published online: 9 April 2010  
© The Author(s) 2010. This article is published with open access at Springerlink.com

**Abstract** Reliable predictions of immunogenic peptides are essential in rational vaccine design and can minimize the experimental effort needed to identify epitopes. In this work, we describe a pan-specific major histocompatibility complex (MHC) class I epitope predictor, *NetCTLpan*. The method integrates predictions of proteasomal cleavage, transporter associated with antigen processing (TAP) transport efficiency, and MHC class I binding affinity into a MHC class I pathway likelihood score and is an improved and extended version of *NetCTL*. The *NetCTLpan* method performs predictions for all MHC class I molecules with known protein sequence and allows predictions for 8-, 9-, 10-, and 11-mer peptides. In order to meet the need for a low false positive rate, the method is optimized to achieve high specificity. The method was trained and validated on large datasets of experimentally identified MHC class I ligands and cytotoxic T lymphocyte (CTL) epitopes. It has been reported that MHC molecules are differentially dependent on TAP transport and proteasomal cleavage. Here, we did not find any consistent signs of such MHC dependencies, and the *NetCTLpan* method is implemented with fixed weights for proteasomal cleavage and TAP transport for all MHC molecules. The predictive performance of the *NetCTLpan* method was shown to outperform other state-of-the-art CTL epitope prediction methods. Our

results further confirm the importance of using full-type human leukocyte antigen restriction information when identifying MHC class I epitopes. Using the *NetCTLpan* method, the experimental effort to identify 90% of new epitopes can be reduced by 15% and 40%, respectively, when compared to the *NetMHCpan* and *NetCTL* methods. The method and benchmark datasets are available at <http://www.cbs.dtu.dk/services/NetCTLpan/>.

**Keywords** MHC class I pathway · HLA · Pan-specific prediction · CTL epitope · MHC polymorphism

## Introduction

Cytotoxic T lymphocytes (CTLs) are a subgroup of T cells able to induce cell death of other cells. CTLs kill only infected or otherwise damaged cells. In order to discriminate between infected and healthy cells, all nucleated cells present host cell peptide fragments on the cell surface in complex with major histocompatibility complex class I molecules (MHC class I). Not all possible peptides originating from cell proteins will be presented by MHC class I. In fact, it is estimated that only one out of 2,000 potential peptides will be immunodominant (Yewdell and Bennink 1999). One of the first steps involved in MHC class I antigen presentation is the degradation of intracellular proteins, including proteins from the cytoplasm and nucleus, by the proteasome (Larsen et al. 2007; Paz et al. 1999; Craiu et al. 1997; Altuvia and Margalit 2000; Mo et al. 1999; Stoltz et al. 1998; Juncker et al. 2009). These peptides may be trimmed at the N-terminal end by cytosolic exopeptidases (Lévy et al. 2002). A subset of the peptides is transported by transporter associated with antigen processing (TAP)

**Electronic supplementary material** The online version of this article (doi:10.1007/s00251-010-0441-4) contains supplementary material, which is available to authorized users.

T. Stranzl (✉) · M. V. Larsen · C. Lundegaard · M. Nielsen  
Department of Systems Biology DTU, Building 208, Center for  
Biological Sequence Analysis, Technical University of Denmark,  
Lyngby 2800, Denmark  
e-mail: stranzl@cbs.dtu.dk

complex into the endoplasmic reticulum (ER), where further N-terminal trimming occurs (Ritz and Seliger 2001; Koch et al. 2004; van Endert et al. 1994; Schatz et al. 2008). Inside the ER, a peptide may bind to an MHC class I molecule and the peptide–MHC complex will be transported to the cell surface, where it subsequently may be recognized by CTLs. These successive steps from protein to ligand presented on the cell surface are limiting the number of possible epitopes. The most restricting step in antigen presentation is peptide binding to MHC class I molecule (Yewdell and Bennink 1999).

Reliable predictions of immunogenic peptides can minimize the experimental effort needed to identify epitopes. We have previously described a method, *NetCTL* (Larsen et al. 2007, 2005), integrating MHC class I binding, TAP transport efficiency, and proteasomal cleavage predictions to an overall prediction of CTL epitopes. The *NetCTL* method has proven successful in identification of CTL epitopes from, for instance influenza (Wang et al. 2007), HIV (Pérez et al. 2008), and *Orthopoxvirus* (Tang et al. 2008). Several other groups have developed methods for CTL epitope identification by integrating steps of the MHC class I pathway (*MAPP*, Hakenberg et al. 2003; *WAPP*, Dönnies and Kohlbacher 2005); *EpiJen*, Doytchinova et al. 2006; *MHC-pathway*, Tenzer et al. 2005). All these methods are limited by the fact that they only allow for prediction of peptide binding to a highly limited set of different MHC molecules. In a large-scale benchmark evaluation of publicly available server of MHC class I pathway presentation prediction, Larsen et al. (2007) showed that the *NetCTL* method significantly outperformed all these methods, closely followed by *MHC-pathway*. The *MHC-pathway* method has recently been updated to include more accurate predictions of MHC binding and a broader allelic coverage (close to 60 human leukocyte antigen (HLA)-A and HLA-B alleles are covered by the default *MHC-pathway* method in the 2009-09-01 release). In contrast to this, the *NetCTL* method has not been updated since 2007, and the MHC binding prediction remains limited to the 12 common HLA supertypes (Lund et al. 2004). In the following, we describe an improved and extended version of *NetCTL*, called *NetCTLpan*, which is able to make predictions for all MHC class I molecules with known protein sequence. In addition, *NetCTLpan* can identify 8-, 9-, 10-, and 11-mer epitopes, as opposed to *NetCTL*, which only allowed for prediction of 9-mer epitopes. The method has been trained on a large data set of experimentally identified MHC ligands from the SYFPEITHI database (Rammensee et al. 1999).

Choosing a performance measure for evaluating a prediction method is a nontrivial task, and the definition of performance measure will often influence the benchmark outcome and subsequent choice of best method. A

commonly used measure for predictive performance is the area under the receiver operating characteristic (ROC) curve, the AUC value. This measure integrates the sensitivity curve as a function of specificity for the range of sensitivity from one to zero. This measure might not be optimal if a prediction method is required to have a very high specificity in order to lower the false positive rate for subsequent experimental validation. In such situations, it could be beneficial to use only the high specificity part of the ROC curve to calculate the predictive performance. To match such requirements for a low false positive rate, we have therefore in this work focused on optimizing the method to achieve high specificity at a potential loss in sensitivity.

The predictive performance of the *NetCTLpan* method is validated on large and MHC diverse data sets derived from the SYFPEITHI (Rammensee et al. 1999) and Los Alamos HIV databases (<http://www.hiv.lanl.gov/>), and its performance has been compared to other state-of-the-art CTL epitope prediction methods.

It has been suggested that supertype-specific differences exist in how dependent MHC class I presentation of peptides is on transport via TAP molecules (Brusic et al. 1999; Anderson et al. 1993; Henderson et al. 1992; Smith and Lutz 1996) and proteasomal cleavage (Wherry et al. 2006). Likewise, it has been suggested that the rescaling procedure commonly used to correct for possible discrepancies between the allelic predictors (Sturniolo et al. 1999; Larsen et al. 2005, 2007) could mask genuine biological difference between MHC molecules and potentially lower the epitope predictive performance (MacNamara et al. 2009). In the context of the *NetCTLpan* method, we investigate to what extent such differences are observed in large data sets that are diverse with regard to both MHC restriction and CTL epitopes.

## Materials

### SYF data set

The SYFPEITHI database (Rammensee et al. 1999) was used as the source of MHC class I ligands. MHC class I binding peptides classified as ligands were downloaded in August 2009. Altogether, the database contained 2,966 HLA class I ligand pairs. Considering only ligands with length of 8 to 11 amino acids (the lengths for which the MHC class I binding predictor *NetMHCpan* can perform predictions), the data set consists of 2,752 unique HLA class I ligand pairs. Data used for training the individual MHC class I pathway predictors—MHC binding (Nielsen et al. 2007; Hoof et al. 2009), proteasomal cleavage (Nielsen et al. 2005), and TAP transport efficiency (Peters

et al. 2003)—was removed from the data set, downsizing it to 2,309 unique HLA class I ligand pairs.

Peptides in the data set with only serotypic HLA assignment were assigned to the most common HLA allele in the European population for this serotype (e.g., the serotype HLA-A\*01 was assigned to the specific allele HLA-A\*0101). The HLA allele frequencies were obtained from the dbMHC database (<http://www.ncbi.nlm.nih.gov/mhc/>). Subsequently, for every peptide, the source protein was found in the UniProtKB/Swiss-Prot database (Uniprot Consortium 2009). If more than one matching protein was a possible source for a peptide, the protein was selected with preference for human and long protein sequences. Peptides without corresponding source protein in UniProtKB/Swiss-Prot were searched against NCBI NR protein database (<http://www.ncbi.nlm.nih.gov>). These steps consequently resulted in the SYF data set consisting of 2,267 HLA class I ligand pairs with corresponding source proteins, where 226 ligands are 8-mers, 1,443 are 9-mers, 430 are 10-mers, and 168 ligands belong to the group of 11-mers. Note, that HLA-C ligands are included in these numbers. In the evaluation, HLA-C ligands are merged to a separate test set.

#### HIV data set

The same HIV data set has been used as for the paper describing the original *NetCTL* method (Larsen et al. 2007). For comparison reasons, the data set has not been updated. The data set is derived from the Los Alamos HIV database (<http://www.hiv.lanl.gov/>). It consists of 216 HLA class I ligand pairs with corresponding source proteins covering the 12 supertypes (Lund et al. 2004).

#### Training and test sets

Each of the HLA alleles in the SYF data set was assigned a supertype association using the distance measure described by Nielsen et al. (2007). In short, an HLA allele was associated to the most similar supertype defined in terms of the correlation coefficient between *NetMHCpan* prediction scores for 1,000,000 random natural 9-mer peptides for the HLA allele in question and any of the 12 supertype representatives (Larsen et al. 2005). In a few cases (less than ten), the supertype association was ambiguous. In these cases, the association was assigned by applying the classification from the work by Sidney et al. (2008). The associated supertypes for each HLA class I allele are shown in Supplementary Table S1. Some supertypes in the 9-mer SYF data set contain more HLA class I ligand pairs than others. Only four out of the 12 supertypes had more than 100 HLA class I ligand pairs assigned. In order to minimize bias toward only a few

supertypes, a training data set with maximum 50 randomly selected ligands per supertype was generated. For seven supertypes, it was possible to select 50 ligands for the training set, while the selection for the five remaining supertypes consisted of between 19 and 47 ligands. This results in a training set of 504 HLA class I ligand pairs. Remaining HLA-A and HLA-B ligands not included in the training data were assigned to a separate set used for evaluation. This evaluation set covers seven supertypes and consists of 889 9-mers. All HLA-A and HLA-B 8-, 10-, and 11-mer ligands were merged into another evaluation set, resulting in a total of 806 ligands. The HIV data set was used as a third independent evaluation set. The numbers of ligands per supertypes for the training and test sets are listed in Table 1. Finally, a set of 65 HLA-C ligands from the SYFPEITHI database of length 8–11 amino acids was used as a fourth evaluation set.

## Methods

### MHC class I affinity prediction

The current version of the pan-specific MHC class I binding prediction method, *NetMHCpan-2.2* (Hoof et al. 2009), is an updated version of the original *NetMHCpan* method (Nielsen et al. 2007). It has been evaluated as the best pan-specific method in large benchmark study (Zhang et al. 2009) and is now including the extension to perform predictions for 8-, 10-, and 11-mer peptides (Lundegaard et al. 2008). *NetMHCpan-2.2* was trained on a data set of 102,146 quantitative peptide–MHC affinity data points covering more than 100 distinct MHC molecules. The prediction server is available at <http://www.cbs.dtu.dk/services/NetMHCpan-2.2/>.

**Table 1** Numbers of ligands per supertype in the training and test sets

| Supertype | Train | Test 9-mer | Test 8-/10-/11-mer | HIV |
|-----------|-------|------------|--------------------|-----|
| A1        | 36    | 0          | 29                 | 5   |
| A2        | 50    | 208        | 94                 | 82  |
| A3        | 50    | 49         | 75                 | 41  |
| A24       | 19    | 0          | 5                  | 9   |
| A26       | 50    | 43         | 74                 | 2   |
| B7        | 50    | 8          | 57                 | 32  |
| B8        | 28    | 0          | 19                 | 5   |
| B62       | 47    | 0          | 27                 | 10  |
| B27       | 50    | 224        | 141                | 3   |
| B39       | 50    | 21         | 36                 | 1   |
| B44       | 50    | 336        | 227                | 16  |
| B58       | 24    | 0          | 22                 | 10  |
| Total     | 504   | 889        | 806                | 216 |

### TAP transport efficiency prediction

The prediction of TAP transport efficiency is based on the matrix method described in Peters et al. (2003). The method predicts TAP transport efficiency of peptides by a scoring method using only the C terminus and the tree N-terminal residues of a peptide. The contribution to the prediction score of the N-terminal residues is down-weighted by a factor of 0.2 in comparison with the score of the C terminus. In the original publication, the TAP transport efficiency score was computed as the average of the values for the 9-mer and its 10-meric precursor. Here, we extend this approach and predict the TAP transport efficiency score for peptides of length from 8 to 11 amino acids, as the average of the values for the original peptide and its precursor extended by one amino acid N-terminally. The matrix published in Peters et al. (2003) was modified as all values in the TAP scoring matrix were multiplied by a factor of  $-1$ , in order to have a high predicted value corresponding to high transport efficiency. This way the interpretation is consistent with the prediction of proteasomal cleavage and MHC class I binding affinity.

### Proteasomal cleavage prediction

*NetChop* C-term 3.0 (Nielsen et al. 2005) was used for predicting cleavage sites. As in the original *NetCTL* publication, only the C-terminal cleavage score of a peptide was included.

### Combined class I pathway presentation prediction—*NetCTLpan*

The *NetCTLpan* prediction value is defined as a weighted sum of the three individual prediction values for MHC class I affinity, TAP transport efficiency, and C-terminal proteasomal cleavage. Optimal relative weights on TAP transport efficiency and proteasomal cleavage were estimated using the training data set and based on the average AUC value per HLA class I ligand pair.

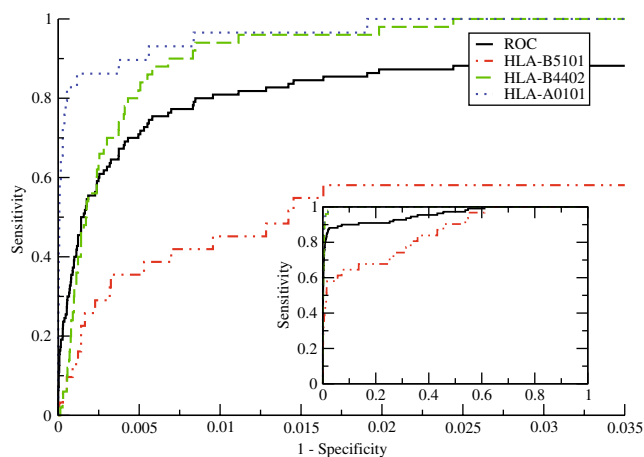
The AUC measure is a commonly used measure for quantitative tests and model comparison. AUC is the area under the ROC curve, summarizing the sensitivity as a function of  $1 - \text{specificity}$ . The specificity is given as  $1 - \text{false positive ratio}$  defined as the fraction of the number of correctly predicted nonligands relative to the total number of nonligands in the dataset (Lund et al. 2005). A specificity of 100% is interpreted as all nonligands are actually classified as nonligands. The sensitivity is the true positive rate (TPR) and is defined as the number of correctly predicted ligands relative to the total number of ligands in the dataset. The higher the TPR, the

more actual positives are recognized. The AUC measure might not be optimal if a prediction method is required to have very high specificity in order to lower the false positive rate in subsequent experimental validations. In such situations, it is beneficial to use only the high specificity part of the ROC curve to calculate the predictive performance. Therefore, a search optimizing the AUC value integrated for specificities from 1 to  $x$  (AUC $_x$ ), where  $x$  [0:1] was performed to optimize the method to achieve high specificity. High values of  $x$  will focus the method toward high specificity at a potential loss in sensitivity, whereas low values of  $x$  will result in equal focus on sensitivity and specificity.

When calculating the AUC value, the source protein was divided into overlapping peptides of the size of the given ligand. All peptides, except those annotated as ligands in either the complete SYFPEITHI or Los Alamos HIV databases, were taken as negative peptides (nonligands) and the given ligand was taken as positive. A perfect AUC value of 1.0 corresponds to the ligand having the highest combined score (*NetCTLpan* score) compared to all other possible peptides originating from the source protein.

Another important issue to resolve is how to calculate AUC values. Should it have been done per protein, where an AUC value is calculated for each ligand–HLA–protein triplet and the performance reported as the average AUC value over all triplets or should it have been made in a pooled way, where all peptide data for the different source proteins and HLA alleles are merged together before calculating the AUC value? Here, we suggest using the per-protein measure, since pooling data from different proteins and HLA alleles will place ligands in a nonbiological competition for presentation. The source proteins in the SYF ligand data sets have a length distribution varying from 36 to more than 8,000 amino acids. Applying the *NetCTLpan* method to our training set (most homogenous data set) shows a tendency for shorter proteins having a lower AUC $_{0.1}$  than longer proteins. Proteins from our training set with length of 0–200 have a mean AUC $_{0.1}$  of 0.817, whereas proteins longer than 200 AA have a mean AUC $_{0.1}$  of 0.876. The Spearman's rank correlation between the protein length and AUC $_{0.1}$  values for the training data set is 0.15. This value is significantly different from random ( $p < 0.001$ , exact permutation test). In a pooled evaluation, where source protein data are merged, the predictive performance would predominantly reflect the performance for the longer protein. Further, not all proteins are expressed in equal amounts within the cell and the presentation of peptides in complex with HLA molecules happens in competition with the four most different HLA-A and HLA-B molecules within a given host and not 46, as it would be the case, when all the HLA

alleles from the SYF training data set are pooled. Finally, it is becoming apparent that not all MHC molecules present peptides at the same binding threshold (Rao et al. 2009). This observation would make an evaluation, where data for different HLA alleles is pooled, highly problematic, as illustrated in Fig. 1. Here, a ROC curve is shown for a pooled set of 29 HLA-A\*0101, 50 HLA-B\*4402, and 31 HLA-B\*5101 ligands using the *NetCTLpan* method. In addition, the allele-specific sensitivity (fraction of ligands identified) for each allele is shown as a function of the pooled specificity. The figure clearly demonstrates that different alleles dominate the ROC curve in different specificity ranges. At a specificity of 0.0025, for instance, 60% (66) of the 110 ligands are identified. Of these are 25 (86% of 29) HLA-A\*0101, 32 (62% of 50) are HLA-B\*4401, and only nine (29% of 31) are HLA-B\*5101 restricted. At very high specificities, the ROC curve is thus predominantly shaped by the HLA-A\*0101 data, at intermediate specificities values the curve is shaped by the HLA-B\*4402 data, and finally at low specificity values, the HLA-B\*5101 data defines the curve. This is clearly not an optimal way of evaluating an overall predictive performance of a prediction method that is aimed at achieving uniform prediction accuracy across a broad range of HLA alleles. To conclude, we find that the proposed triplet evaluation per ligand–HLA–protein evaluation constitutes the least biased approach to evaluate a prediction method with broad allelic coverage.

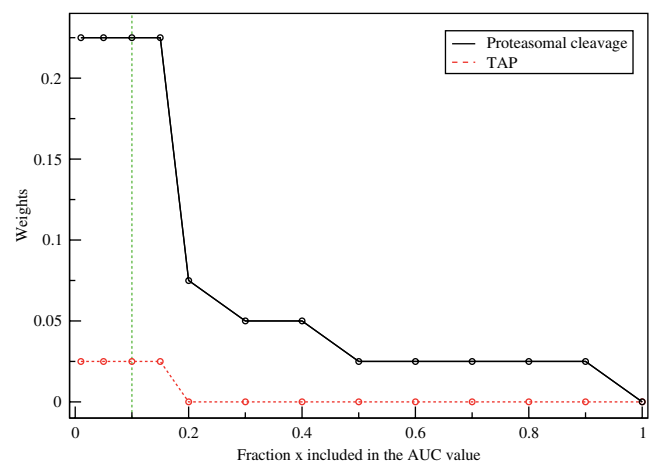


**Fig. 1** ROC curves for a pooled data set from the HLA-A\*0101, HLA-B\*4402, and HLA-B\*5101 alleles. The source proteins for all three alleles were cut into overlapping peptides of the size of the given ligand, and all peptides except the given ligands were taken as negative. The data set contained 31 HLA-A\*0101, 50 HLA-B\*4402, and 29 HLA-B\*5101 ligands, and the predictions were made using the *NetCTLpan* method. The *black curve* shows the ROC curve for the combined data set. The *other three curves* show the allele-specific sensitivity (fraction of ligands identified) as a function of the overall specificity for each of the three alleles. The *insert* shows the curves for the full range of specificities

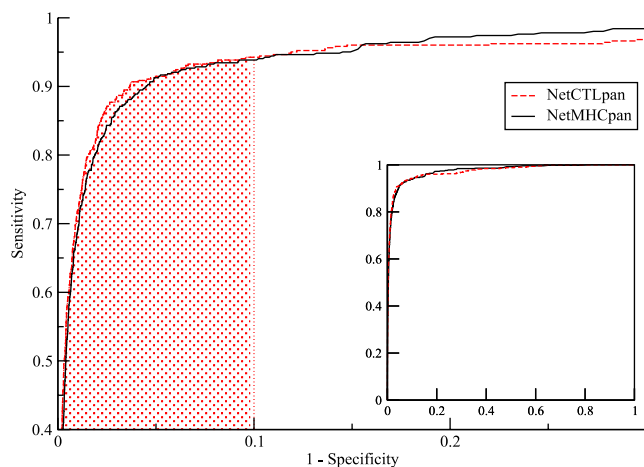
## Results

### The *NetCTLpan* method

The optimal weights on proteasomal cleavage and TAP transport efficiency were calculated for AUC fractions ( $AUC_x$ ) varying  $x$  from 0.05 to 1, with a step size of 0.05. With  $x$  equal to 1, this corresponds to the conventional AUC value calculation and the way of selecting optimal weights for the original *NetCTL* method. The result of this analysis is shown in Fig. 2. For an AUC fraction of 1, the optimal weights were zero on both proteasomal cleavage and TAP transport. This implies that *NetMHCpan 2.2*, the method used for predicting MHC class I binding affinity, has a very high performance and that adding predictions for proteasomal cleavage or TAP transport decreased the overall performance. Figure 2 illustrates that the more the method is focused on high specificity (low values of  $x$ ), the higher the weights and thus importance of proteasomal cleavage and TAP transport predictions become. This is, however, achieved at a loss in sensitivity at low specificity values. Based on this observation, the best performing weights on proteasomal cleavage and TAP transport were selected using an AUC fraction of 0.1 as benchmark measure and were found to be 0.225 for cleavage and 0.025 for TAP. This selection of weights defines the *NetCTLpan* method. When interpreting the weights for cleavage and TAP, keep in mind that the contribution of the different prediction methods is not directly reflecting their relative biological contribution in the pathway.



**Fig. 2** Weights on proteasomal cleavage and TAP transport efficiency related to  $AUC_x$  fraction. The smaller the included fraction, the higher the contribution of proteasomal cleavage and TAP transport efficiency to a high performance. Optimal weights on proteasomal cleavage and TAP were found by optimizing the average  $AUC_x$  value on the SYF training data set. The *dotted line* indicates the  $AUC_{0.1}$  fraction



**Fig. 3** Performance comparison in terms of ROC curves for *NetCTLpan* and *NetMHCpan*. The true positive rate is shown as a function of the false positive rate. The figure is based on the SYF training set. The shaded area shows the area under the curve used to calculate the  $AUC_{0.1}$ . The insert shows the complete curves

A comparison of the ROC curves for *NetMHCpan* and our described method *NetCTLpan* is shown in Fig. 3. The overall AUC value for the *NetMHCpan* method is 0.980 and the corresponding  $AUC_{0.1}$  value is 0.852. For the *NetCTLpan* method, the overall AUC value is 0.976 and the corresponding  $AUC_{0.1}$  value is 0.869. These numbers and the graphs in Fig. 3 illustrate the improved specificity of the *NetCTLpan* method compared to *NetMHCpan*. Up to a specificity of 0.85, the ROC curve for *NetCTLpan* has a higher sensitivity than *NetMHCpan*, indicating that this method will identify more true ligands at a given specificity threshold. On the other hand, below a specificity of 0.85, the two ROC curves cross and the *NetMHCpan* method achieves the highest sensitivity. This crossover, however, happens at a very low specificity corresponding to a false positive rate of 0.15 (15% of the negative peptides are falsely classified as positive) and is of limited use when doing actual epitope discovery work, underlining the importance of optimizing the methods on high specificity.

Table 2 displays the comparison between *NetCTLpan* and *NetMHCpan* for the different data sets using both the overall AUC and  $AUC_{0.1}$  benchmark measures. Using the  $AUC_{0.1}$  measure, the *NetCTLpan* method has a significantly higher performance compared to *NetMHCpan* for all data sets. On the other hand, when comparing the overall AUC value, the two methods show comparable performance. Here, for the SYF data set, the *NetMHCpan* method has the highest performance, while for the HIV data set and the HLA-C test set, *NetCTLpan* performs best. So, if high sensitivity is essential (even at a cost in specificity), the *NetMHCpan* method should be preferred. In more common situations, where specificity is the more important issue, *NetCTLpan* should be the choice.

Results displayed in Table 2 are mean AUC and  $AUC_{0.1}$  values over all ligand–HLA–protein triplets in each data set. Paired tests were used for comparing performance between different prediction methods. In Supplementary Table S2 are given the AUC and  $AUC_{0.1}$  values for each ligand–HLA–protein triple in the SYFPEITHI data sets. From this table, it is clear that the predictive performance does not only vary between supertypes, but also within supertypes. For the training data set, the difference between HLA-B\*5101 and HLA-B\*0702 (both B7 supertype alleles) for the *NetCTLpan* method is thus 0.374 in terms of the  $AUC_{0.1}$  measure. These performance variations demonstrate the need for large-scale HLA diverse benchmark data set to evaluate differences in performance between prediction methods, as the performance difference between similar (supertype-wise) alleles often is as high as the difference for individual alleles between two prediction methods within a given data set.

#### Data redundancy

Several ligands appear in the SYFPEITHI ligand data sets as duplicates restricted to multiple HLA class I alleles. One might be worried that the potential peptide

**Table 2** AUC and fractional AUC value comparison between *NetCTLpan* and *NetMHCpan*

| Data             | Measure     | NetCTLpan    | NetMHCpan    | <i>p</i> value |
|------------------|-------------|--------------|--------------|----------------|
| Train (9)        | AUC         | 0.976        | <b>0.980</b> | 0.056          |
|                  | $AUC_{0.1}$ | <b>0.869</b> | 0.852        | 0.002          |
| Test (8/9/10/11) | AUC         | 0.977        | <b>0.979</b> | 0.273          |
|                  | $AUC_{0.1}$ | <b>0.863</b> | 0.855        | 0.002          |
| Test (HIV)       | AUC         | <b>0.933</b> | 0.920        | 0.028          |
|                  | $AUC_{0.1}$ | <b>0.612</b> | 0.593        | 0.106          |
| Test (HLA-C)     | AUC         | <b>0.920</b> | 0.866        | <0.001         |
|                  | $AUC_{0.1}$ | <b>0.495</b> | 0.307        | <0.001         |

The performance values are calculated as average per protein AUC values over the corresponding data sets. *p* values are calculated by a paired *t* test excluding ties. The best performing method is, for each data set and performance measure, highlighted in bold

**Table 3** Supertype-specific weights benchmark

| Supertype | Weights  |       | Train |              |                | Test (8/9/10/11) |              |                | Test (HIV)   |              |                |
|-----------|----------|-------|-------|--------------|----------------|------------------|--------------|----------------|--------------|--------------|----------------|
|           | Cleavage | TAP   | Fixed | Specific     | <i>p</i> value | Fixed            | Specific     | <i>p</i> value | Fixed        | Specific     | <i>p</i> value |
| A1        | 0.050    | 0.075 | 0.942 | <b>0.950</b> | 0.294          | <b>0.937</b>     | 0.936        | 0.326          | 0.381        | <b>0.455</b> | 0.610          |
| A2        | 0.550    | 0.000 | 0.808 | <b>0.822</b> | 0.133          | <b>0.776</b>     | 0.758        | 0.008          | <b>0.681</b> | 0.657        | 0.104          |
| A3        | 0.225    | 0.025 | 0.890 | 0.890        | 0.598          | 0.872            | 0.872        | (a)            | 0.648        | 0.648        | (a)            |
| A24       | 0.000    | 0.000 | 0.917 | <b>0.942</b> | 0.257          | 0.783            | <b>0.895</b> | 0.389          | 0.636        | <b>0.636</b> | 0.960          |
| A26       | 0.275    | 0.025 | 0.885 | <b>0.885</b> | 0.476          | <b>0.873</b>     | 0.864        | 0.006          | 0.761        | <b>0.771</b> | 0.500          |
| B7        | 0.000    | 0.000 | 0.710 | <b>0.736</b> | 0.378          | 0.765            | <b>0.765</b> | 0.998          | <b>0.493</b> | 0.437        | 0.064          |
| B8        | 0.725    | 0.000 | 0.916 | <b>0.920</b> | 0.231          | 0.858            | <b>0.870</b> | 0.517          | 0.132        | <b>0.144</b> | 0.374          |
| B62       | 0.275    | 0.200 | 0.889 | <b>0.902</b> | 0.014          | <b>0.751</b>     | 0.727        | 0.345          | 0.440        | <b>0.496</b> | 0.303          |
| B27       | 0.475    | 0.025 | 0.911 | <b>0.921</b> | 0.014          | <b>0.921</b>     | 0.902        | 0.001          | <b>0.370</b> | 0.299        | 0.390          |
| B39       | 0.175    | 0.025 | 0.859 | <b>0.860</b> | 0.896          | <b>0.853</b>     | 0.849        | 0.362          | <b>0.739</b> | 0.711        | (b)            |
| B44       | 0.100    | 0.025 | 0.859 | <b>0.868</b> | 0.127          | 0.885            | <b>0.896</b> | 0.001          | <b>0.636</b> | 0.631        | 0.845          |
| B58       | 0.025    | 0.025 | 0.959 | <b>0.963</b> | 0.399          | 0.820            | <b>0.887</b> | 0.161          | 0.774        | <b>0.882</b> | 0.128          |
| All       | 0.225    | 0.025 | 0.869 | <b>0.878</b> | 0.016          | <b>0.863</b>     | 0.860        | 0.143          | <b>0.612</b> | 0.603        | 0.300          |

Optimal weights per supertype are shown. Performance is given as the average  $AUC_{0.1}$  value for each data set. Fixed weights for proteasomal cleavage and TAP transport efficiency are 0.225 and 0.025, respectively. The higher  $AUC_{0.1}$  value is highlighted in bold for each data set and supertype

(a)  $AUC_{0.1}$  values are equal for fixed and specific weights, (b) only one sample available for the given supertype

similarity/redundancy could influence the performance estimates of the *NetCTLpan* method. The training data set, for instance, consists of 504 HLA ligand pairs, but only 492 of these are unique peptides. The 9-mer test set consists of 889 9mer HLA ligand pairs, of which 802 are unique peptides. The training and 9-mer test sets share 42 identical ligands and three ligands with one mismatch, all coupled to different alleles. The training set contains four ligands identical with one mismatch. To investigate the impact on this data redundancy within the training data set and between the training and test data sets, we calculated the performance on redundancy-reduced data sets. The performance on the training set was calculated by removing duplicates and ligands with one mismatch and for the test set by excluding duplicates and ligands with one mismatch to ligands in the training data. Predictive performance was shown to be close to identical for both training and test set, suggesting that peptide redundancy plays a negligible role in our performance evaluation (see data in Supplementary Table S3).

#### MHC affinity rescaling

In contrast to the *NetCTL* method, the *NetCTLpan* method does not use rescaling of predicted MHC class I affinities. Previously, rescaling has been used to make prediction values comparable between MHC class I molecules. It has been suggested that such a rescaling might remove genuine biological differences between

MHC molecules and potentially lowers the epitopes predictive performance (MacNamara et al. 2009). To investigate, if the predictive performance of the *NetCTLpan* method is influenced when including rescaling, we defined a rescaling factor for each MHC allele and used that factor to rescale all MHC binding affinity values before integrating with proteasomal cleavage and TAP scores. For each allele, the rescaling factor was determined as the 1 percentile score of the *NetMHCpan* method for a set of 1,000,000 random natural 9-mer peptides. An overall performance gain using rescaling as compared to not applying rescaling was observed if focusing on the overall AUC value (no rescaling AUC 0.976 versus rescaling AUC 0.978, *p* value 0.006, paired *t* test). For high specificity predictions ( $AUC_{0.1}$ ), however, the method without rescaling performed similar ( $AUC_{0.1}$  0.869) to the method using rescaling ( $AUC_{0.1}$  0.868) with a *p* value of 0.835. From these results, and to maintain potential biological differences in specificity between MHC molecules, we chose not to include rescaling in the *NetCTLpan* method. One might argue that rescaling versus non-rescaling cannot influence the performance of the *NetCTLpan* method, when the performance is calculated per ligand–HLA allele, as it is the case in this study. When focusing on MHC binding predictions alone, this is true and both methods give identical results. However, when integrated with proteasomal cleavage and TAP transport efficiency, this situation changes. Rescaling places all MHC binding predictions on a similar scale and hence

also places the relative weights on TAP and proteasomal cleavage on a similar scale across the set of MHC alleles. This is no longer the case if rescaling is left out. Here, alleles with low (predicted) binding affinity preference will have higher relative weights on TAP and proteasomal cleavage as compared to alleles with high binding affinity preference.

#### Supertype-specific weights on proteasomal cleavage and TAP scores

As mentioned earlier, previous work has suggested that different MHC molecules have different dependencies on TAP transport efficiency and proteasomal cleavage. Based on these observations, it seems natural to find allele-specific weights for TAP transport and proteasomal cleavage. Due to the small size of the training data set, we limited ourselves to a search for supertype-specific weights. For each supertype, we estimated the weights on proteasomal cleavage and TAP transport that give optimal average  $AUC_{0.1}$  values. Optimal weights per supertype and performance values for the different data sets can be seen in Table 3. It shows that relative large differences exist between the optimal weights across the different superotypes. Naturally, the average  $AUC_{0.1}$  for the training set is higher with supertype-specific weights as compared to the fixed weights (estimate for the complete training data set). Applying these weights resulted in an inconsistent pattern in performance gain across the different superotypes for the different test sets when compared to fixed weights. Only three superotypes (A24, B8, and B58) showed a consistent performance gain for the SYFPHITHI and HIV test sets using supertype-specific weights. This result strongly indicates that optimal weights per supertype are not reflecting biological differences but occur most likely due to overfitting. Note that we are not stating that proteasomal cleavage and TAP transport dependency could not vary between MHC molecules; we only state that based on our data, we cannot consistently reproduce such a differentiated dependency.

#### Comparison to *NetCTL*

The comparison of the performance between *NetCTLpan* and *NetCTL* is based on the 9-mer data sets, since *NetCTL* is only capable of predicting 9-meric epitopes. Table 4 shows the performance for *NetCTLpan* and *NetCTL* on the different data sets. For both SYF data sets, the *NetCTLpan* method significantly outperforms *NetCTL*. The HIV test set does not show *NetCTLpan* being significantly better than *NetCTL*. The HIV test set is supertype based, and the HLA restriction for each HIV epitope is assigned to the

**Table 4** Benchmark comparison of the *NetCTLpan* and the *NetCTL* methods

| Data       | Measure     | <i>NetCTLpan</i> | <i>NetCTL</i> | <i>p</i> value     |
|------------|-------------|------------------|---------------|--------------------|
| Train (9)  | AUC         | <b>0.976</b>     | 0.971         | 0.018              |
|            | $AUC_{0.1}$ | <b>0.869</b>     | 0.816         | <0.001             |
| Test (9)   | AUC         | <b>0.982</b>     | 0.975         | <0.001             |
|            | $AUC_{0.1}$ | <b>0.877</b>     | 0.802         | <0.001             |
| Test (HIV) | AUC         | 0.933            | <b>0.936</b>  | 0.366 <sup>a</sup> |
|            | $AUC_{0.1}$ | <b>0.612</b>     | 0.606         | 0.600              |

Average AUC and  $AUC_{0.1}$  values for the *NetCTLpan* and *NetCTL* methods calculated for the SYF train set and the SYF and HIV test sets. For each data set and performance measure, the best performing method is shown in bold. *p* values are calculated by a paired *t* test excluding ties

<sup>a</sup> When using full HLA typing information, the *NetCTLpan* performance values are 0.959 and 0.745 for AUC and  $AUC_{0.1}$ , respectively. Both these values are significantly higher than the values of *NetCTL*

corresponding HLA supertype. This is in contrast to the SYF ligand data sets, where full typing HLA restriction is available for most ligands. One hundred nineteen out of 216 HIV peptide supertype pairs are, however, annotated in the Los Alamos HIV database with full typing for the HLA restriction. Using this additional information about the HLA restriction improves the mean  $AUC_{0.1}$  from 0.612 to 0.745 and the overall AUC from 0.933 to 0.959. Both measurements thus testify *NetCTLpan* as having a significantly better performance (both *p* values <0.001, paired *t* test) compared to *NetCTL*. These results clearly confirm earlier findings (Pérez et al. 2008; Hoof et al. 2009) of the importance of going beyond HLA superotypes and the use of full-type HLA restriction information when identifying MHC class I epitopes.

To determine the source of the strong gain in predictive performance between the *NetCTL* and *NetCTLpan* methods, we compared the predictive performance of the *NetCTLpan* method to that of *NetCTL* using the supertype representative for each HLA allele also for the *NetCTLpan* method. This analysis clearly shows (see Table 5) that the shift from supertype to allele-specific predictions is the main driving force behind the gain in predictive performance between *NetCTL* and *NetCTLpan*. In all benchmarks has the *NetCTLpan*\_ST (supertype-specific *NetCTLpan* method) a similar predictive performance to that of *NetCTL*.

#### Comparison to state-of-the-art MHC class I pathway prediction methods

Next, we compared the performance of the *NetCTLpan* method to the *MHC-pathway* method (Tenzer et al. 2005). This method has earlier been shown to be a state-of-the-art MHC class I pathway predictor (Larsen et al. 2005). Like the *NetCTLpan* method, this method integrates predictions of



**Table 5** Benchmark comparison of *NetCTL*, *NetCTLpan*, and *NetMHCpan\_ST* (supertype-specific version of *NetCTLpan*)

| Data           | Measure            | <i>NetCTL</i> | <i>NetCTLpan</i> | <i>NetCTLpan_ST</i> |
|----------------|--------------------|---------------|------------------|---------------------|
| Train (9)      | AUC                | 0.971         | 0.976            | 0.971               |
|                | AUC <sub>0.1</sub> | 0.816         | 0.869            | 0.830               |
| Test (9)       | AUC                | 0.975         | 0.982            | 0.971               |
|                | AUC <sub>0.1</sub> | 0.802         | 0.877            | 0.805               |
| Test (8/10/11) | AUC                | NA            | 0.972            | 0.961               |
|                | AUC <sub>0.1</sub> | NA            | 0.848            | 0.770               |

The performance values are calculated as average per protein AUC values for the training and test data sets

MHC binding, C-terminal proteasomal cleavage, and TAP transport into a combined pathway presentation score. Here, we use the method with default parameters via the link [http://tools.immuneepitope.org/analyze/html/mhc\\_processing.html](http://tools.immuneepitope.org/analyze/html/mhc_processing.html). The *MHC-pathway* method is not pan-specific and hence does not allow predictions for all HLA class I alleles used in our benchmark data. Further, it does not allow for predictions of 8- and 11-mer epitopes and only allows 10-mer epitope predictions for a subset of the included alleles. To allow for a fair comparison, we therefore only included ligands from the SYF data set restricted to HLA alleles covered by the *MHC-pathway* method. The results of the benchmark calculation are shown in Table 6 and clearly show that *NetCTLpan* outperforms the *MHC-pathway* method for all three data sets. The improved performance is maintained for both the AUC and AUC<sub>0.1</sub> measure. Further, the table shows that the MHC binding predictors for the two methods have close to identical performance (*NetMHCpan* versus *MHC*). The cleavage method employed by the *NetCTLpan* method is performing consistently better than the immunoproteasome prediction method used by *MHC-pathway* (*NetChop* versus *Immu*). The TAP prediction method is identical between the two methods. These results suggest that the integration method employed by *MHC-pathway* is not optimal either due to the relative low performance of the immunoproteasome predictor or as a consequence of how the three

prediction scores have been integrated in the *MHC-pathway* method.

**Discussion**

Earlier work has demonstrated the benefit of integrating proteasomal cleavage, TAP transport efficiency, and MHC binding predictions when using reverse immunology to identify potential CTL epitopes. However, to the best of our knowledge, none of the publicly available methods providing this integration are pan-specific and hence do not allow for prediction of CTL epitopes restricted to any MHC allele.

Here, we have developed a pan-specific MHC class I epitope predictor, *NetCTLpan*. The method integrates prediction of proteasomal cleavage, TAP transport efficiency, and MHC binding into a MHC class I pathway presentation likelihood score. In large-scale benchmarks comprising more than 1,000 MHC class I ligands and CTL epitopes restricted by close to 60 different HLA alleles, the method was shown to outperform both the original *NetCTL* method, as well as *MHC-pathway*, another state-of-the-art class I presentation pathway prediction method.

*NetCTLpan* was optimized to achieve high specificity in order to meet the need for a low false positive rate when

**Table 6** Benchmark comparison of the *NetCTLpan* and *MHC-pathway* methods

| Data      | Measure            | <i>NetCTLpan</i> | <i>MHC-pathway</i> | <i>p</i> value | <i>NetMHCpan</i> | <i>MHC</i> <sup>a</sup> | <i>TAP</i> | <i>NetChop</i> | <i>Immu</i> <sup>b</sup> | <i>N</i> |
|-----------|--------------------|------------------|--------------------|----------------|------------------|-------------------------|------------|----------------|--------------------------|----------|
| Train (9) | AUC                | 0.978            | 0.972              | <0.001         | 0.983            | 0.981                   | 0.839      | 0.881          | 0.803                    | 438      |
|           | AUC <sub>0.1</sub> | 0.874            | 0.854              | 0.01           | 0.858            | 0.862                   | 0.278      | 0.360          | 0.260                    | 438      |
| Test (9)  | AUC                | 0.978            | 0.974              | <0.001         | 0.978            | 0.977                   | 0.809      | 0.870          | 0.774                    | 615      |
|           | AUC <sub>0.1</sub> | 0.871            | 0.847              | <0.001         | 0.864            | 0.870                   | 0.204      | 0.362          | 0.215                    | 615      |
| Test (10) | AUC                | 0.966            | 0.957              | <0.005         | 0.964            | 0.966                   | 0.810      | 0.817          | 0.734                    | 291      |
|           | AUC <sub>0.1</sub> | 0.842            | 0.800              | <0.005         | 0.835            | 0.824                   | 0.272      | 0.238          | 0.180                    | 291      |

The performance values are calculated as average per protein AUC values for the training and test data sets. The benchmark is made on the subset of the SYF ligand data sets covered by the *MHC-pathway* method

<sup>a</sup> MHC prediction score from *MHC-pathway* method

<sup>b</sup> Immunoproteasomal cleavage score from *MHC-pathway* predictions. The TAP prediction method is identical between the two methods. *p* value for the comparison of *NetCTLpan* to *MHC-pathway* are calculated by a paired *t* test excluding ties

using the method for large-scale epitope discovery. If focusing on optimal sensitivity, it was shown that the optimal prediction method should exclude both cleavage and TAP predictions reducing the method to MHC binding prediction alone. This is in contrast to earlier work, where proteasomal cleavage and TAP transport efficiency consistently have been reported to improve the predictive performance. Whether this observation reflects true biological aspects of the specificity overlap between the three pathway players (see for instance Nielsen et al. 2005) or it simply occurs because the prediction of MHC class I affinity has gained accuracy during the recent years, whereas predictors for TAP transport efficiency and proteasomal cleavage have not changed or been updated, remains to be seen.

Recent publications have suggested that some MHC molecules are, compared to others, more or less dependent on TAP transport and proteasomal cleavage. Using the *NetCTLpan* method in large-scale benchmarks, we however find no consistent signal of such an HLA allele differentiated dependency of proteasomal cleavage and TAP transport efficiency. A performance gain using supertype-specific weights could only be observed for the training set. Applying these weights to the test sets resulted in an inconsistent pattern in performance gain for the different superotypes when compared to fixed weights, indicating that optimal weights per supertype are not reflecting biological differences but most likely are a result of overfitting.

*NetCTL*, the ancestor of *NetCTLpan*, uses a rescaling of MHC binding affinity values to make prediction values comparable between MHC class I molecules. It has been suggested that such a rescaling might remove genuine biological differences between MHC molecules and potentially lower the method's predictive performance. Here, we show that rescaling has no significant impact on the overall predictive performance of the *NetCTLpan* method. Further, we observed a tendency of different MHC molecules presenting ligands at different (predicted) binding thresholds. Based on these observations, the *NetCTLpan* method is implemented without use of rescaling, thus maintaining potential genuine biological differences between MHC molecules. To allow comparison between presentation likelihood scores for different MHC molecules, we include a rank-score for each prediction. The rank-score is calculated as the percent rank of a given *NetCTLpan* likelihood score to a set of 200,000 random natural 9-mer peptides.

Our results on the HIV benchmark data set confirm the importance of going beyond HLA superotypes and use full-type HLA restriction information when identifying MHC class I epitopes. In this benchmark, we found a significantly improved predictive performance, if full HLA restriction were used, in comparison to the HLA supertype information proposed in the original *NetCTL* publication.

In contrast to earlier published methods for MHC class I pathway prediction, *NetCTLpan* allows for predictions of 8- to 11-mer CTL epitopes being presented by any MHC class I molecule of known protein sequence.

*NetCTLpan*, the method described in this work, has shown to perform best when focusing on high specificity predictions for CTL epitope identification. In order to easily grasp the predictive performance gain, we applied the rank measure as defined by Larsen et al. (2005). The rank measure reports the average fraction of epitopes identified as a function of the percentage rank (percentage of tested peptides) for a set of proteins. This measure indicates how large a fraction of the peptides for a given protein needs to be tested in order to identify the epitope with a given likelihood. To identify new epitopes with 90% likelihood by use of *NetCTLpan*, the rank measure reports that 3.7% of the peptides need to be experimentally verified. For a hypothetical protein of 300 peptides, this means that on average, 11 peptides need to be tested in order to identify the epitope. The corresponding numbers for *NetMHCpan* and *NetCTL* are 13 and 17 peptides. Hence, by applying the *NetCTLpan* method instead of *NetMHCpan*, the experimental effort can be reduced by 17%, and compared to *NetCTL*, approximately 40% fewer epitopes need to be tested. Based on this, it is clear that utilizing the *NetCTLpan* method can minimize experimental effort needed to identify new CTL epitopes. We believe that this improved performance, combined with the method's ability to provide predictions of potential CTL epitopes of length from 8 to 11 amino acids to any MHC class I molecules of known sequence, will be useful in both rational reverse immunogenetic epitope discovery and interpretation of observed immune responses in HLA diverse patient cohorts. The *NetCTLpan* method and benchmark data set are available at: <http://www.cbs.dtu.dk/services/NetCTLpan>.

**Acknowledgments** This work is supported by a grant from the Danish Research Council for Technology and Production Sciences (project title "Disease Gene Finding, Somatic Mutations, and Vaccine Design," principal funding recipient is Søren Brunak) and by NIH (National Institute of Health) grants (contract no. HHSN266200400083C, principal funding recipient is Ole Lund; contract no. HHSN266200400025C, principal funding recipient is Søren Buus; contract no. HHSN266200400006C, principal funding recipient is Alessandro Sette).

**Open Access** This article is distributed under the terms of the Creative Commons Attribution Noncommercial License which permits any noncommercial use, distribution, and reproduction in any medium, provided the original author(s) and source are credited.

## References

- Altuvia Y, Margalit H (2000) Sequence signals for generation of antigenic peptides by the proteasome: implications for proteasomal cleavage mechanism. *J Mol Biol* 295:879–890

- Anderson KS, Alexander J, Wei M, Cresswell P (1993) Intracellular transport of class I MHC molecules in antigen processing mutant cell lines. *J Immunol* 151:3407–3419
- Brusic V, van Endert P, Zeleznikow J, Daniel S, Hammer J, Petrovsky N (1999) A neural network model approach to the study of human TAP transporter. *In Silico Biol* 1:109–121
- Craiu A, Akopian T, Goldberg A, Rock KL (1997) Two distinct proteolytic processes in the generation of a major histocompatibility complex class I-presented peptide. *Proc Natl Acad Sci USA* 94:10850–10855
- Doytchinova I, Guan P, Flower D (2006) EpiJen: a server for multistep T cell epitope prediction. *BMC Bioinformatics* 7:131
- Dönnes P, Kohlbacher O (2005) Integrated modeling of the major events in the MHC class I antigen processing pathway. *Protein Sci* 14:2132–2140
- Hakenberg J, Nussbaum A, Schild H, Rammensee H, Kuttler C, Holzhütter H, Kloetzel P, Kaufmann S, Mollenkopf H (2003) MAPPP: MHC class I antigenic peptide processing prediction. *Appl Bioinformatics* 2:155–158
- Henderson RA, Michel H, Sakaguchi K, Shabanowitz J, Appella E, Hunt DF, Engelhard VH (1992) HLA-A2.1-associated peptides from a mutant cell line: a second pathway of antigen presentation. *Science* 255:1264–1266
- Hoof I, Peters B, Sidney J, Pedersen LE, Sette A, Lund O, Buus S, Nielsen M (2009) NetMHCpan, a method for MHC class I binding prediction beyond humans. *Immunogenetics* 61:1–13
- Juncker AS, Larsen MV, Weinhold N, Nielsen M, Brunak S, Lund O (2009) Systematic characterisation of cellular localisation and expression profiles of proteins containing MHC ligands. *PLoS ONE* 4:e7448
- Koch J, Guntrum R, Heintke S, Kyritsis C, Tampé R (2004) Functional dissection of the transmembrane domains of the transporter associated with antigen processing (TAP). *J Biol Chem* 279:10142–10147
- 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
- Larsen MV, Lundegaard C, Lamberth K, Buus S, Lund O, Nielsen M (2007) Large-scale validation of methods for cytotoxic T-lymphocyte epitope prediction. *BMC Bioinformatics* 8:424
- Lévy F, Burri L, Morel S, Peitrequin A, Lévy N, Bachi A, Hellman U, Van Den Eynde BJ, Servis C (2002) The final N-terminal trimming of a subaminoterminal proline-containing HLA class I-restricted antigenic peptide in the cytosol is mediated by two peptidases. *J Immunol* 169:4161–4171
- Lund O, Nielsen M, Kesmir C, Petersen AG, Lundegaard C, Worning P, Sylvester-Hvid C, Lamberth K, Røder G, Justesen S, Buus S, Brunak S (2004) Definition of supertypes for HLA molecules using clustering of specificity matrices. *Immunogenetics* 55:797–810
- Lund O, Nielsen M, Lundegaard C, Kesmir C, Brunak S (2005) *Immunological bioinformatics*, 1 ed. MIT press
- Lundegaard C, Lund O, Nielsen M (2008) Accurate approximation method for prediction of class I MHC affinities for peptides of length 8, 10 and 11 using prediction tools trained on 9mers. *Bioinformatics* 24:1397–1398
- MacNamara A, Kadolsky U, Bangham CR, Asquith B (2009) T-cell epitope prediction: rescaling can mask biological variation between MHC molecules. *PLoS Comput Biol* 5:e1000327
- Mo XY, Cascio P, Lemerise K, Goldberg AL, Rock K (1999) Distinct proteolytic processes generate the C and N termini of MHC class I-binding peptides. *J Immunol* 163:5851–5859
- Nielsen M, Lundegaard C, Lund O, Kesmir C (2005) The role of the proteasome in generating cytotoxic T-cell epitopes: insights obtained from improved predictions of proteasomal cleavage. *Immunogenetics* 57:33–41
- Nielsen M, Lundegaard C, Blicher T, Lamberth K, Harndahl M, Justesen S, Røder G, Peters B, Sette A, Lund O, Buus S (2007) NetMHCpan, a method for quantitative predictions of peptide binding to any HLA-A and -B locus protein of known sequence. *PLoS ONE* 2:e796
- Paz P, Brouwenstijn N, Perry R, Shastri N (1999) Discrete proteolytic intermediates in the MHC class I antigen processing pathway and MHC I-dependent peptide trimming in the ER. *Immunity* 11:241–251
- Pérez CL, Larsen MV, Gustafsson R, Norström MM, Atlas A, Nixon DF, Nielsen M, Lund O, Karlsson AC (2008) Broadly immunogenic HLA class I supertype-restricted elite CTL epitopes recognized in a diverse population infected with different HIV-1 subtypes. *J Immunol* 180:5092–5100
- Peters B, Bulik S, Tampe R, Van Endert PM, Holzhütter H (2003) Identifying MHC class I epitopes by predicting the TAP transport efficiency of epitope precursors. *J Immunol* 171:1741–1749
- Rammensee H, Bachmann J, Emmerich NP, Bachor Oa, Stevanović S (1999) SYFPEITHI: database for MHC ligands and peptide motifs. *Immunogenetics* 50:213–219
- Rao X, Costa AI, van Baarle D, Kesmir C (2009) A comparative study of HLA binding affinity and ligand diversity: implications for generating immunodominant CD8+ T cell responses. *J Immunol* 182:1526–1532
- Ritz U, Seliger B (2001) The transporter associated with antigen processing (TAP): structural integrity, expression, function, and its clinical relevance. *Mol Med* 7:149–158
- Schatz MM, Peters B, Akkad N, Ullrich N, Martinez AN, Carroll O, Bulik S, Rammensee H, van Endert P, Holzhütter H, Tenzer S, Schild H (2008) Characterizing the N-terminal processing motif of MHC class I ligands. *J Immunol* 180:3210–3217
- Sidney J, Peters B, Frahm N, Brander C, Sette A (2008) HLA class I supertypes: a revised and updated classification. *BMC Immunology* 9:1
- Smith KD, Lutz CT (1996) Peptide-dependent expression of HLA-B7 on antigen processing-deficient T2 cells. *J Immunol* 156:3755–3764
- Stoltze L, Dick TP, Deeg M, Pömmnerl B, Rammensee HG, Schild H (1998) Generation of the vesicular stomatitis virus nucleoprotein cytotoxic T lymphocyte epitope requires proteasome-dependent and -independent proteolytic activities. *Eur J Immunol* 28:4029–4036
- Sturniolo T, Bono E, Ding J, Radrizzani 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
- Tang ST, Wang M, Lamberth K, Harndahl M, Dziegiel MH, Claesson MH, Buus S, Lund O (2008) MHC-I-restricted epitopes conserved among variola and other related orthopoxviruses are recognized by T cells 30 years after vaccination. *Arch Virol* 153:1833–1844
- Tenzer S, Peters B, Bulik S, Schoor O, Lemmel C, Schatz MM, Kloetzel P, Rammensee H, Schild H, Holzhütter H (2005) Modeling the MHC class I pathway by combining predictions of proteasomal cleavage, TAP transport and MHC class I binding. *Cell Mol Life Sci: CMLS* 62:1025–1037
- Uniprot consortium (2009) The universal protein resource (UniProt) 2009. *Nucleic Acids Res* 37:169–174
- Wang M, Lamberth K, Harndahl M, Røder G, Stryhn A, Larsen MV, Nielsen M, Lundegaard C, Tang ST, Dziegiel MH, Rosenkvist J, Pedersen AE, Buus S, Claesson MH, Lund O (2007) CTL epitopes for influenza A including the H5N1 bird flu; genome-, pathogen-, and HLA-wide screening. *Vaccine* 25:2823–2831

- Wherry EJ, Golovina TN, Morrison SE, Sinnathamby G, McElhaugh MJ, Shockey DC, Eisenlohr LC (2006) Re-evaluating the generation of a “proteasome-independent” MHC class I-restricted CD8 T cell epitope. *J Immunol* 61:2249
- Yewdell JW, Bennink JR (1999) Immunodominance in major histocompatibility complex class I-restricted T lymphocyte responses. *Annu Rev Immunol* 17:51–88
- van Endert PM, Tampé R, Meyer TH, Tisch R, Bach JF, McDevitt HO (1994) A sequential model for peptide binding and transport by the transporters associated with antigen processing. *Immunity* 1:491–500
- Zhang H, Lundegaard C, Nielsen M (2009) Pan-specific MHC class I predictors: a benchmark of HLA class I pan-specific prediction methods. *Bioinformatics* 25:83–89