

A stochastic π -calculus model of MHC class I antigen presentation

Leonard Goldstein

19th August 2005

Contents

1	Introduction	2
2	Background	2
3	Modelling	5
3.1	Basic model	5
3.1.1	Stochastic π -calculus implementation	5
3.1.2	Model parameters and discussion	8
3.1.3	Simulation results	10
3.2	Flytrap model	13
3.2.1	Stochastic π -calculus implementation	13
3.2.2	Model parameters and discussion	17
3.2.3	Simulation results	19
3.3	Time-dependent optimization of peptide repertoire	20
3.3.1	Simulation results	20
3.3.2	Discussion	21
3.4	Model testing	23
3.4.1	MHC class I heavy chain alleles	23
3.4.2	Model fitting	24
4	Conclusion	27
5	Acknowledgements	27
A	SPiM source code	28
A.1	Basic model	28
A.2	Flytrap model (a)	30
A.3	Flytrap model (b)	32

1 Introduction

The aim of this project is to create a stochastic π -calculus model of MHC class I antigen presentation, in order to test proposed molecular mechanisms underlying MHC class I assembly.

Stochastic π -calculus is a stochastic variant of π -calculus [1], which can be viewed as a programming language for the modular description of concurrent non-deterministic systems. Components of a system are described as processes, which precisely define possible actions and interactions with other system components. Processes communicate with each other through channels, which fire at stochastic rates. A non-deterministic choice essentially translates to a race between exponentially distributed waiting times. Thus stochastic π -calculus can be used for precise and concise description of non-deterministic systems. Furthermore, it can be used for model testing, as any model described in the language can be executed by simulators based on the Gillespie algorithm [2]. Traditionally, process calculi have mostly been used as a theoretical framework with little practical application. Recently, they have gained practical importance for applications in biology, where the compositional approach is exploited to directly describe biological systems. For this project, simulations are run on the Stochastic Pi Machine (SPiM), a formally specified simulator, which has been proved correct with respect to the calculus [3].

The modelled events in MHC class I antigen presentation play an important role in the cellular immune response. At the cell surface, MHC class I molecules present peptides derived from intracellular or extracellular protein to CD8⁺ T cells, which recognize self and foreign peptides. They are also involved in the regulation of natural killer cells and in the development of self tolerant T cells in the thymus [4]. Peptide loading of the MHC complex occurs in the endoplasmic reticulum (ER) and is assisted by multiple cofactors. Optimal peptide selection ensures sufficient stability of the MHC complex both for trafficking to and maintaining at the cell surface. Many aspects of MHC class I assembly are still poorly understood and are the subject of on-going research.

2 Background

The following overview of MHC class I assembly is based on review articles [5, 6, 7].

The MHC class I complex is a ternary complex of MHC class I heavy chain (HC), beta-2 microglobulin (β_2m) and peptide, which is formed through non-covalent interactions. MHC class I associated peptides are 8 to 11 residues in length (typically 9). Degradation of ubiquitin tagged proteins by the proteasome within the cytosol generates peptides consisting of 4 to 20 amino acids. Thus peptides frequently have to undergo trimming by aminopeptidases, either in the cytoplasm or ER, prior to binding to an HC/ β_2m complex. The different cofactors involved in MHC class I assembly are briefly introduced below.

Transporter associated with antigen processing (TAP) TAP is an ER-resident heterodimeric peptide transporter, which translocates peptides from the cytosol into the lumen of the ER in an ATP-dependent manner. Transported

peptides are optimally 9-12 amino acids in length, although longer peptides may be transported.

Tapasin Tapasin bridges HC/ β_2m to TAP. In addition, it stabilizes TAP and facilitates the incorporation of calreticulin and ERp57 to the multimolecular peptide loading complex at the TAP interface.

Calnexin/Calreticulin Calnexin and Calreticulin are homologous chaperones, which assist in the folding of newly synthesized glycoproteins in the ER. They contribute to ER quality control to ensure that only correctly folded and assembled proteins exit the ER.

ERp57 ERp57 is an oxidoreductase containing two thioredoxin-like (TR-like) motifs, which are active sites of disulfide bond reduction/oxidation. Mediation of disulfide bond formation often occurs in association with calnexin or calreticulin.

MHC class I assembly can be divided in three different phases. Early events lead to the formation of a folded and fully oxidized HC/ β_2m complex, which is peptide receptive (Figure 1). Later events result in the acquisition of peptide which provides sufficient stability for the HC/ β_2m /peptide trimer. Finally, late events describe the exit of the loaded MHC complex from the ER and their transport to the cell surface.

Early events In absence of β_2m , the MHC class I heavy chain binds to calnexin, which is believed to promote efficient protein folding. Two conserved intrachain disulfide bonds are essential for correct assembly and are likely to be formed in presence of ERp57. ERp57 binds to calnexin with low affinity, allowing for rapid dissociation after the disulfide bonds are formed. Calnexin-deficient cells still exhibit MHC class I surface expression, and it has been suggested that calnexin could be functionally replaced. A likely candidate is the immunoglobulin binding protein (BiP). Upon β_2m association, calnexin is usually replaced by calreticulin.

Later events Peptide acquisition occurs within the so-called peptide loading complex (PLC) (Figure 1). It consists of the MHC class I heavy chain, β_2m , the general-purpose molecules calreticulin and ERp57 and the MHC class I specific molecules TAP and tapasin. It has been established that tapasin bridges TAP to the HC/ β_2m complex. The presence of ERp57 suggests the alteration of disulfide bonds. Surprisingly, residue Cys57 in the TR1 motif of ERp57 was found to form a disulfide bond with Cys95 in tapasin. At the same time, the TR2 motif appears to remain available to undergo redox reactions and may therefore act on the MHC complex. The covalent bond to tapasin could serve the purpose of retaining ERp57 in the PLC, given that it only exhibits low binding affinity to calreticulin. It has been proposed that the HC/ β_2m complex initially binds suboptimal peptide, which is subsequently replaced by optimal peptide in the PLC. As heavy chain, β_2m and peptide bind cooperatively, optimal peptide stabilizes the MHC heterotrimer. MHC class I peptide cargo has been shown to be optimized both qualitatively and quantitatively over time in

presence of tapasin. Even though tapasin is assumed to be the main component of the PLC, the complex fails to operate at full efficiency without all of its cofactors. Interestingly, different MHC class I heavy chain alleles exhibit a spectrum of dependence upon the PLC. It has been suggested that the PLC originally evolved to achieve optimal loading with a broad repertoire of stable complexes which are sufficient to initiate an immune response. After pathogens adapted to counter the improved pathway, MHC class I alleles could have diversified, resulting in less PLC dependent alleles [4].

Late events Upon dissociation from TAP, the MHC complex exits from the ER and enters the golgi apparatus through the ER golgi intermediate compartment. Peptide binding promotes unbinding from TAP, while the rate of transport to the golgi apparatus remains unchanged. MHC complexes are believed to cluster at ER exit sites and associate with the putative transport receptor BAP31. Cells deficient in tapasin or TAP show no egresssion of truly empty molecules, suggesting that in absence of suitable peptide cargo, peptide receptive MHC complexes are actively detained.

Degradation Misfolded MHC class I molecules are segregated from the folding environment in the ER. After disulfide bond reduction and unfolding, they are retrotranslocated from the ER into the cytosol, where they are degraded by proteasome.

More detailed results on individual aspects of peptide loading are introduced in the context of modelling in section 3.

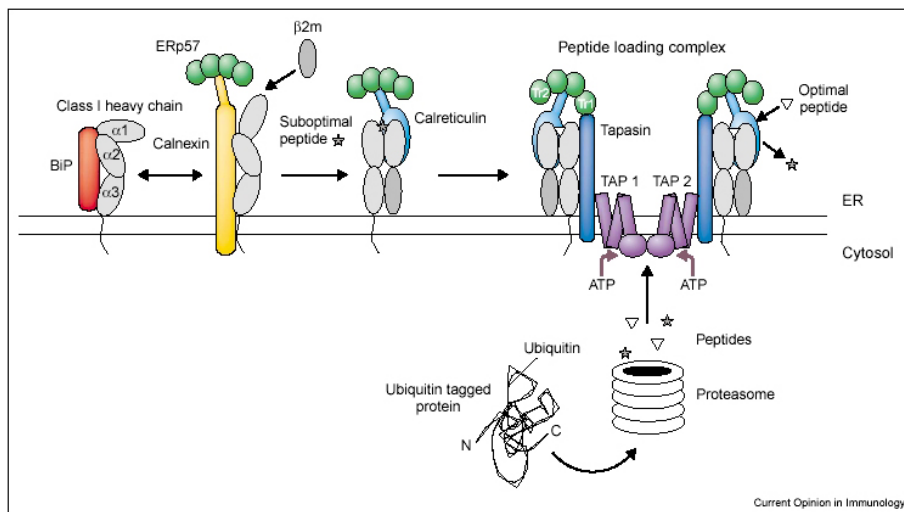


Figure 1: Formation of the peptide loading complex (see [5])

3 Modelling

A number of general assumptions are made for the purpose of modelling.

The HC/ β_2 m dimer is treated as one unit. As peptide and β_2 m bind cooperatively to the heavy chain, peptide binding to the HC/ β_2 m dimer is favoured compared to binding to free HC. Hence it is generally assumed that at low peptide concentrations in the ER, the complex of β_2 m and HC forms first, and peptide then binds to the HC/ β_2 m complex [8]. Moreover, β_2 m dissociation from the peptide loaded complex does not depend on the binding properties of the bound peptide [6]. Hence it has no effect on the composition of the peptide repertoire of loaded MHC complexes. β_2 m dissociation can therefore simply be interpreted as a possible form of degradation.

Consider three different categories of peptide with comparable association rate but different dissociation rates (T. Elliott, personal communication). The three corresponding HC/ β_2 m/peptide complexes exhibit low, medium and high equilibrium association constants K_a and show half times of approximately 100, 250 and 600 minutes respectively. In the following, assume dissociation rates of 3.0, 1.2 and 0.5. The model does not allow for peptide dissociation after the time point of egression from the ER, as this would imply a greater proportion of MHC complexes loaded with high affinity peptide at the cell surface than were originally egressed, thus complicating the analysis of events in the ER.

Three main aspects of peptide loading can be identified as

- (1) peptide selection in favour of peptide with low dissociation rate
- (2) total number of loaded MHC complexes in a given time
- (3) time-dependent optimization of previously acquired peptide cargo

The above points differ between MHC class I heavy chain alleles and are affected by tapasin. Initially, it is assumed that all aspects of peptide loading are poor in absence of tapasin and are improved in its presence.

Qualitative (1) and quantitative (2) aspects of peptide loading are discussed in 3.1 and 3.2. Time-dependent peptide optimization (3) is addressed in 3.3, and section 3.4 introduces more specific MHC class I heavy chain alleles.

3.1 Basic model

3.1.1 Stochastic π -calculus implementation

A basic model for MHC peptide loading is illustrated in Figure 2. Each component in the reaction diagram is defined as a stochastic π -calculus process (for correspondence, see Table 1). To introduce general concepts of the SPiM Language, the source code for this model is presented in more detail than in the following sections. The main text only contains selected parts of the code, while the complete code for every model is given in the Appendix. Further details on SPiM can be found in the SPiM User Manual [9].

Processes `generate_MHC` and `generate_pep` generate HC/ β_2 m and peptide at rates `dg_MHC` and `dg_pep` respectively. After a stochastic delay, they rein-

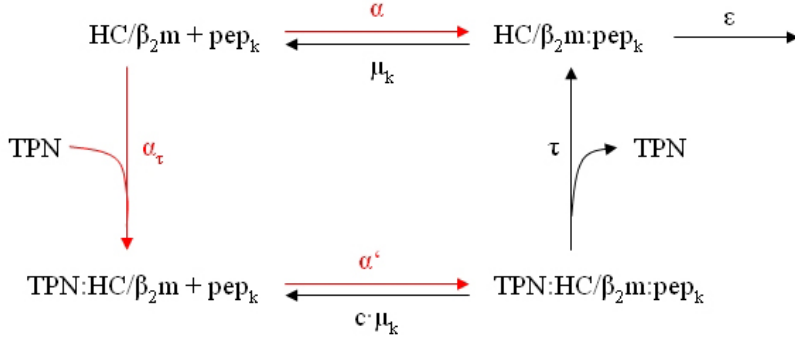


Figure 2: Basic model. Red arrows indicate reactions through public channels, black arrows represent reactions modelled by stochastic delays (followed by communication through a private channel at infinite rate if required). Letters next to arrows denote rate constants. ϵ represents the rate of egression from the ER.

HC/ β_2 m	MHC
pep	pep
HC/ β_2 m:pep	MHC_pep
TPN	TPN
TPN:HC/ β_2 m	MHC_TPN
TPN:HC/ β_2 m:pep	MHC_pep_TPN

Table 1: Modelled components and corresponding SPiM processes

state themselves, spawning off a process MHC or pep at the same time (parallel composition is denoted by |).

Consider the process pep. The argument mu can take values mu_low, mu_med or mu_high, corresponding to low, medium and high affinity (or high, medium, low dissociation rate). The remaining arguments r0, r1, r2 are zero rate channels which are used to track the state of a given peptide (free, bound or egressed). For example, !f1, !b2, !e3 correspond to free low K_a peptide, bound medium K_a peptide and egressed high K_a peptide respectively.

pep can bind to MHC through a public channel a_pep, by passing on its dissociation rate mu, along with two private channels u and e. If there is a parallel input ?a_pep to output !a_pep on channel a_pep, processes MHC and pep cease to exist and processes MHC_pep and pep_bound are instantiated with the given values. As binding is reversible, there are still two separate processes corresponding to HC/ β_2 m and peptide. The two complementary processes MHC_pep and pep_bound communicate events of unbinding or egression through private channels u and e respectively. Communication over u reinstates the original processes MHC and pep. Channel privacy guarantees that pep_bound can only dissociate from the one MHC_pep it was bound to in the first place. After egression, that is parallel output !e and input ?e over channel e, MHC_pep and pep_bound cease to exist, and the process pep_egressed is instantiated. As HC/ β_2 m/peptide is considered stable after egression, the complex is now treated as a single unit. It is more convenient to spawn this process from within pep_bound rather than

```

let generate_MHC()=
  delay@dg_MHC;(generate_MHC()|MHC())

let generate_pep()=
  do delay@dg_pep;(generate_pep()|pep(mu_low,f1,b1,e1))
  or delay@dg_pep;(generate_pep()|pep(mu_med,f2,b2,e2))
  or delay@dg_pep;(generate_pep()|pep(mu_high,f3,b3,e3))

let MHC()=
  do ?a_pep(mu,u,e);MHC_pep(mu,u,e)
  or ?a_TPN(u_TPN);MHC_TPN(u_TPN)
  or delay@degrad_MHC

and MHC_pep(mu:float,u:chan,e:chan)=
  do delay@mu;!u;MHC()
  or delay@epsilon;!e

and MHC_TPN(u_TPN:chan)=
  ?a_pep_TPN(mu,u,e);MHC_pep_TPN(mu,u,e,u_TPN)

and MHC_pep_TPN(mu:float,u:chan,e:chan,u_TPN:chan)=
  do delay@mu*c_TPN;!u;MHC_TPN(u_TPN)
  or delay@tau;!u_TPN;MHC_pep(mu,u,e)

let TPN()= (
  new u_TPN:chan
  !a_TPN(u_TPN);(?u_TPN;TPN())
)

let pep(mu:float,r0:chan,r1:chan,r2:chan)= (
  new u:chan new e:chan
  do !r0
  or !a_pep(mu,u,e);pep_bound(mu,u,e,r0,r1,r2)
  or !a_pep_TPN(mu,u,e);pep_bound(mu,u,e,r0,r1,r2)
  or delay@degrad_pep
)

and pep_bound(mu:float,u:chan,e:chan,r0:chan,r1:chan,r2:chan) =
  do !r1
  or ?u;pep(mu,r0,r1,r2)
  or ?e;pep_egressed(r2)

and pep_egressed(r2:chan) =
  do !r2
  or delay@degrad_MHC

```

from `MHC_pep`, as the peptide process has been instantiated with representation channels corresponding to its binding properties (see process `generate_pep`). This information can easily be passed on to `pep_egressed` in the form of channel `r2`, while it is unavailable within the process `MHC_pep`.

A binding reaction always follows the scheme described above. Binding occurs after communication through a public channel, which can be used to send any information required by the complementary process or pass on any private channel required for the communication between the two bound processes. At least one private channel is necessary for unbinding. Additional private channels become necessary if the bound complex can undergo other transitions which affect both components and thus require synchronization between them.

Similar to binding to MHC through channel `a_pep`, `pep` can bind to `MHC_TPN` via channel `a_pep_TPN`. Indeed, for any process `pep`, there is a choice between binding to MHC, `MHC_TPN` and a stochastic delay at rate `degrad_pep`, which corresponds to degradation (not shown in diagram). The three peptide processes `pep`, `pep_bound` and `pep_egressed` offer outputs on channels `r0`, `r1` and `r2` respectively. As mentioned above, these outputs are used for plotting purposes only, and there is no communication on the channels that would affect the model.

For any process `MHC`, there is a choice between binding to `pep`, binding to `TPN` (with unbinding channel `u_TPN`) or degradation at rate `degrad_MHC` (not shown in diagram). `MHC_TPN` only performs one action, namely binding `pep`. It cannot dissociate from tapasin in absence of peptide.

`MHC_pep` dissociates peptide by sending output `!u` after a delay at rate `mu` or egresses by sending output `!e` after a delay at rate `epsilon`. Similarly, `MHC_pep_TPN` dissociates peptide by sending `!u` after a delay at rate `c_TPN*mu` or unbinds from tapasin by sending `!u_TPN` after a delay at rate `tau`. Note that the private channels `u` and `e` have infinite rate, and reaction rates are determined by stochastic delays prior to communicating through the infinitely fast channels. This is essential in the case of peptide dissociation, as peptide unbinding rates differ between `MHC_pep` and `MHC_pep_TPN`. Once `pep` has bound to `MHC_TPN`, the resulting `MHC_pep_TPN` can unbind from `TPN` to become `MHC_pep`, but the common private channel `u` with process `pep_bound` stays the same. Thus the same `u` is used to communicate peptide unbinding both in the case of `MHC_pep_TPN` and `MHC_pep`. Hence an unbinding channel with finite rate could not achieve the desired dissociation rates `mu` and `c_TPN*mu` for `MHC_pep` and `MHC_pep_TPN` respectively.

3.1.2 Model parameters and discussion

Transitions modelled as a delay action, or as the output of a private channel with single corresponding input, occur after an exponentially distributed time with constant rate. Hence they are equivalent to transitions in a continuous-time Markov chain. The analysis of parameters can thus be based on the continuous-time Markov chain in Figure 3.

Let \mathbb{P}_k and \mathbb{P}'_k be conditional probabilities of transitions from `HC/β2m/pepk` with equilibrium constant $K_a = k$ in absence and presence of tapasin respec-

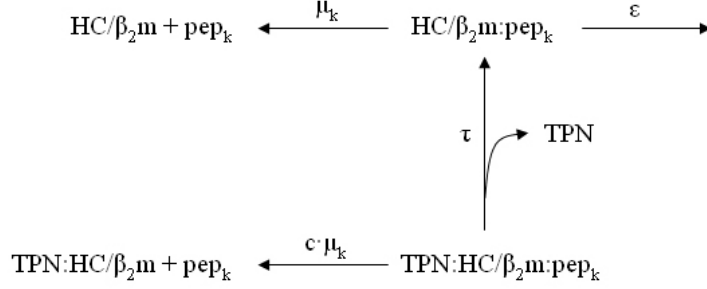


Figure 3: Continuous-time Markov chain as implied by the reaction scheme in Figure 2

tively. Moreover, define events D = peptide dissociation, E = egression. Then

$$\begin{aligned} \mathbb{P}_k(D) &= \frac{\mu_k}{\mu_k + \epsilon} & \mathbb{P}'_k(D) &= \frac{c \cdot \mu_k}{c \cdot \mu_k + \tau} \\ \mathbb{P}_k(E) &= \frac{\epsilon}{\mu_k + \epsilon} & \mathbb{P}'_k(E) &= \frac{\tau}{c \cdot \mu_k + \tau} \frac{\epsilon}{c \cdot \mu_k + \epsilon} \end{aligned}$$

A necessary condition for improved peptide selection in presence of tapasin is given by

$$\mathbb{P}_{low}(E) \approx \mathbb{P}_{med}(E) \approx \mathbb{P}_{high}(E)$$

$$\mathbb{P}'_{low}(E) \ll \mathbb{P}'_{med}(E) \ll \mathbb{P}'_{high}(E).$$

Alternatively, consider the conditional distribution of egressed $\text{HC}/\beta_2\text{m}/\text{peptide}$ complexes. In absence of tapasin, can assume a uniform prior $\mathbb{P}(K_a = k) = \frac{1}{3}$ for all k , for $\text{HC}/\beta_2\text{m}$ is equally likely to bind to low, medium or high K_a peptide. Using Bayes' Theorem, obtain

$$\mathbb{P}(K_a = k|E) = \frac{\mathbb{P}_k(E)}{\sum_j \mathbb{P}_j(E)} = \left(\frac{\mu_k + \epsilon}{\mu_{low} + \epsilon} + \frac{\mu_k + \epsilon}{\mu_{med} + \epsilon} + \frac{\mu_k + \epsilon}{\mu_{high} + \epsilon} \right)^{-1}$$

The analogous distribution for peptide loading in presence of tapasin is more complex, as tapasin affects the prior distribution of K_a . The distribution of $\text{HC}/\beta_2\text{m}/\text{pep}_k$ is no longer the result of a binding reaction between $\text{HC}/\beta_2\text{m}$ and pep_k alone (which is equally likely to occur for $k=\text{low}$, medium or high), but $\text{HC}/\beta_2\text{m}/\text{pep}_k$ also enters the non-tapasin pathway by dissociating from tapasin. These MHC complexes will be biased towards high K_a by assumption. Instead, consider the simpler conditional distribution of $\text{HC}/\beta_2\text{m}/\text{peptide}$ dissociated from tapasin

$$\mathbb{P}'(K_a = k|\text{TPN unbinding}) = \left(\frac{\mu'_k + \tau}{\mu'_{low} + \tau} + \frac{\mu'_k + \tau}{\mu'_{med} + \tau} + \frac{\mu'_k + \tau}{\mu'_{high} + \tau} \right)^{-1}$$

Let T be the event of tapasin unbinding. To model improved peptide selection, now need

$$\mathbb{P}(\text{low}|E) \approx \mathbb{P}(\text{med}|E) \approx \mathbb{P}(\text{high}|E)$$

$$\mathbb{P}'(\text{low}|T) \ll \mathbb{P}'(\text{med}|T) \ll \mathbb{P}'(\text{high}|T).$$

It is clear that egression must be sufficiently fast compared to peptide dissociation. Indeed, as $\epsilon \rightarrow \infty$, $\mathbb{P}(K_a = k|E) \rightarrow \frac{1}{3}$ for all k . On the other hand, τ must be sufficiently small compared to $c \cdot \mu_k$ to allow for peptide discrimination through tapasin. This can be achieved by small values of τ or large values of c (or a combination of both). Note that the two alternatives are formally equivalent with regard to \mathbb{P}'_τ . For $\tau \rightarrow 0$, or equivalently, $c \rightarrow \infty$, obtain a limit distribution (0.105, 0.263, 0.632).

Note that the above probability calculations critically depend on the stated assumptions. The analysis can be helpful for understanding the effects and dependencies of parameters, but it fails under more subtle conditions. A qualitative discussion of events under less restrictive assumptions is given in the following.

Peptide selection is always achieved by increasing the number of times a tapasin-bound HC/ β_2 m binds and unbinds peptide before egression. In absence of tapasin, HC/ β_2 m/peptide is egressed shortly after peptide binding, and no peptide discrimination takes place. The increased number of bound peptide before egression in presence of tapasin can be achieved by retention of HC/ β_2 m (slow tapasin unbinding) or by acceleration of peptide dissociation.

Even though MHC retention and increased peptide dissociation have the same discriminative effect in terms of \mathbb{P}'_τ , there are differences due to the interdependence of tapasin and non-tapasin loading. In general, if tapasin-mediated peptide loading is inefficient with regard to quantity, peptide loading will be shifted towards the non-tapasin pathway. A great proportion of egressed MHC complexes will then be loaded in absence of tapasin, and peptide discrimination is no longer observed. It is obvious that peptide selection therefore requires a sufficiently large number of tapasin molecules. Moreover, parameters for generation and degradation of HC/ β_2 m and peptide have to be chosen carefully to avoid high peptide or HC/ β_2 m concentrations. Due to the limited number of tapasin molecules, excess peptide or HC/ β_2 m always lead to a shift towards the non-tapasin pathway. At the same time, peptide selection requires sufficiently large peptide turnover to ensure that enough high K_a peptides are available when low, medium and high K_a peptide are produced at the same rate.

Consider the implications of slow tapasin unbinding and accelerated peptide dissociation. If tapasin unbinding is too slow, or equivalently, tapasin is saturated with HC/ β_2 m, clearly more MHC complexes will be loaded in absence of tapasin. Interestingly, a similar effect is observed for fast peptide dissociation. Now tapasin is not a limiting factor, but peptide dissociation is so fast that only few HC/ β_2 m/peptide complexes last long enough to be dissociated from tapasin. Again, more MHC complexes will be loaded in absence of tapasin. Both effects are somewhat reflected in the conditional probabilities for egression $\mathbb{P}_k(E) = \frac{\epsilon}{\mu_k + \epsilon}$ and $\mathbb{P}'_k(E) = \frac{\tau}{c \cdot \mu_k + \tau} \frac{\epsilon}{\mu_k + \epsilon}$. For tapasin to be effective, must have $\mathbb{P}'_k(E) \ll \mathbb{P}_k(E)$, at least for $k = \text{high}$. This corresponds to a lower limit for τ and an upper limit for c . There is a tradeoff between optimal peptide selection and efficient loading.

Note that neither slow tapasin unbinding nor accelerated peptide dissociation alone can account for an increase in the total number of loaded MHC complexes in presence of tapasin. Assuming similar peptide association rate constants α and α' for both pathways, peptide loading in presence of tapasin will match the level of peptide loading in absence of tapasin. Increased overall

peptide loading can only be achieved if peptide binding to HC/ β_2 m is favoured in presence of tapasin ($\alpha' > \alpha$). Possible biological interpretations include the prevention of reversible denaturation of the peptide binding site [8] or an equilibrium shift towards a peptide-receptive conformation [4]. In addition, tapasin binding to TAP ensures proximity to the source of translocated peptide. However, experiments with soluble tapasin constructs suggest that the effect is not related to peptide abundance as soluble tapasin is able to largely restore peptide acquisition [10].

3.1.3 Simulation results

Each plot in Figure 4-7 shows a single representative run of the SPiM simulator. The number of low, medium and high K_a HC/ β_2 m/peptide complexes at the cell surface is plotted against time in blue, pink and yellow respectively. HC/ β_2 m is generated at rate `dg_MHC=10.0`. Low, medium and high affinity peptide are all generated at rate `dg_pep=20.0` and degraded at rate `degrad_pep=10.0` to assure sufficient peptide turnover and avoid excess molecules. Free empty MHC complexes in the ER and egressed loaded MHC complexes at the cell surface are degraded at rate `degrad_MHC=0.1` (regardless of K_a in the latter case). For tapasin simulations, assume 100 tapasin molecules.

Figure 4 illustrates how selection of peptide in absence of tapasin depends on egression.

Figure 5 assumes tapasin-mediated MHC loading only ($\alpha = 0.0$, $\alpha' = 1.0$) to compare the effect of slow tapasin unbinding with accelerated peptide dissociation. For high egression rates ($\epsilon = 1000.0$), peptide dissociation after tapasin unbinding can be neglected, and the plots reflect the distribution of loaded MHC complexes dissociated from tapasin \mathbb{P}'_τ . In half of the simulations, tapasin is assumed to have a retention function (top), while the other half assumes accelerated peptide dissociation (bottom). The two alternative modelled functions show similar discriminative power. Reduced overall peptide loading is observed both for strong retention of HC/ β_2 m ($\tau = 0.1$ and $\tau = 0.01$) and for extreme acceleration of peptide dissociation ($c = 10000.0$), the latter being more efficient with regard to quantity. Note how peptide selection improves from left to right (with decreasing τ or increasing c), while overall loading is reduced. This illustrates the tradeoff between optimal peptide selection and loading efficiency.

Figure 6 assumes similar peptide affinity for free HC/ β_2 m and HC/ β_2 m bound to tapasin ($\alpha = \alpha' = 1.0$). Thus MHC loading now occurs in presence or absence of tapasin. For efficient tapasin loading (first, second and third column), the results are similar to Figure 5. For inefficient tapasin-mediated loading (right column), a shift towards the non-tapasin pathway can be observed, resulting in less optimal peptide selection. Note that the overall level of MHC loading stays constant for all parameter combinations.

Figure 7 illustrates how an increased peptide association rate constant for tapasin-bound MHC can account for increased overall peptide loading in presence of tapasin.

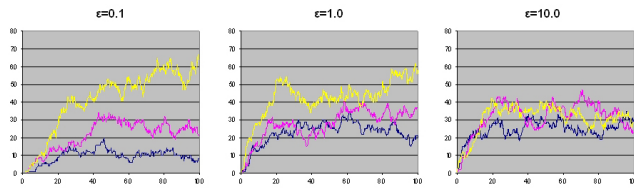


Figure 4: MHC loading in absence of tapasin ($\alpha=1.0$)

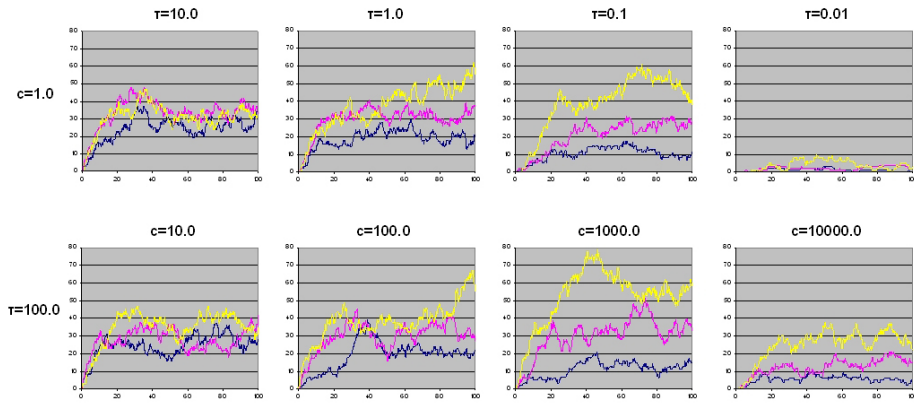


Figure 5: MHC loading through tapasin only ($\alpha=0.0$, $\alpha'=1.0$, $\alpha_\tau=1.0$, $\epsilon=1000.0$)

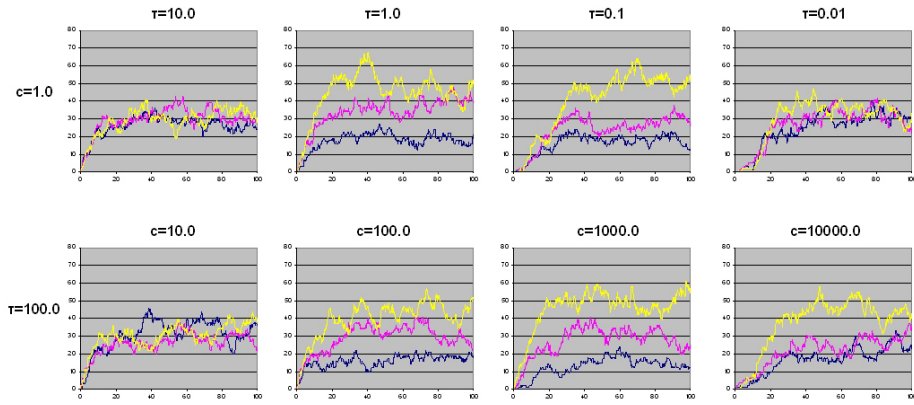


Figure 6: MHC loading in presence of tapasin ($\alpha=1.0$, $\alpha'=1.0$, $\alpha_\tau=1.0$, $\epsilon=1000.0$)

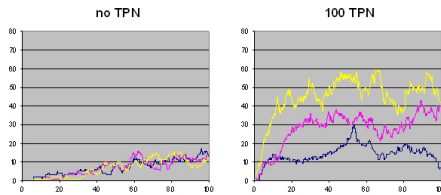


Figure 7: MHC loading in presence and absence of tapasin ($\alpha=0.001$, $\alpha'=1.0$, $\alpha_\tau=1.0$, $\epsilon=100.0$, $c=100.0$ $\tau=10.0$)

3.2 Flytrap model

The basic model described under 3.1 allows modelling of qualitative and quantitative improvement of peptide loading in presence of tapasin. However, peptide binding is thought to be a more complex process.

Analysing in vitro binding of HC/ β_2 m and peptide, Springer *et al.* provided experimental evidence for a conformational change in MHC class I molecules upon peptide binding [8]. They determined off-rates k_{off} and equilibrium association constants K_a for a range of different peptides and found that the predicted association rates $K_a \cdot k_{off}$ were considerably smaller than the observed on-rates. From these kinetic mismatches of up to three orders of magnitude, they inferred that peptide binding might occur in several steps, most likely via a conformational rearrangement of the peptide binding groove.

These results lead Elliott and Williams to propose a "venus fly trap" model [6, 4]. An initial encounter between peptide and HC/ β_2 m could induce a conformational change which would capture the ligand if its rate of dissociation from the encounter complex were slower than the rate of conformational change [4].

3.2.1 Stochastic π -calculus implementation

The flytrap model illustrated in Figure 8 and 9 is based on [6]. When peptide has bound to HC/ β_2 m, there are three possible outcomes: Peptide dissociation without inducing a conformational change, peptide dissociation after inducing conformational change, leaving the MHC complex closed and empty, and peptide capture.

In absence of tapasin, the reaction scheme reduces to the non-tapasin pathway in Figure 8. In presence of tapasin, the tapasin pathway (Figure 9) becomes available and peptide loading will occur through both pathways. Note that only empty HC/ β_2 m can bind to tapasin and thus enter the tapasin pathway. Once bound to tapasin, empty HC/ β_2 m cannot unbind, while loaded HC/ β_2 m/peptide is immediately dissociated when molecule closure is complete. This is consistent with the reported promotion of TAP unbinding in presence of peptide [5].

For the remainder of this section, MHC reopening ($\text{pep}_k : \text{MHC}_{\text{closed}} \longrightarrow \text{pep}_k : \text{MHC}_{\text{open}}$) is ignored. Time-dependent optimization of peptide cargo is discussed in 3.3.

The flytrap concept gives rise to a number of possible implementations. Two equivalent alternatives (a) and (b) are presented here to illustrate the functioning of the stochastic π -calculus simulator. An obvious difference between the

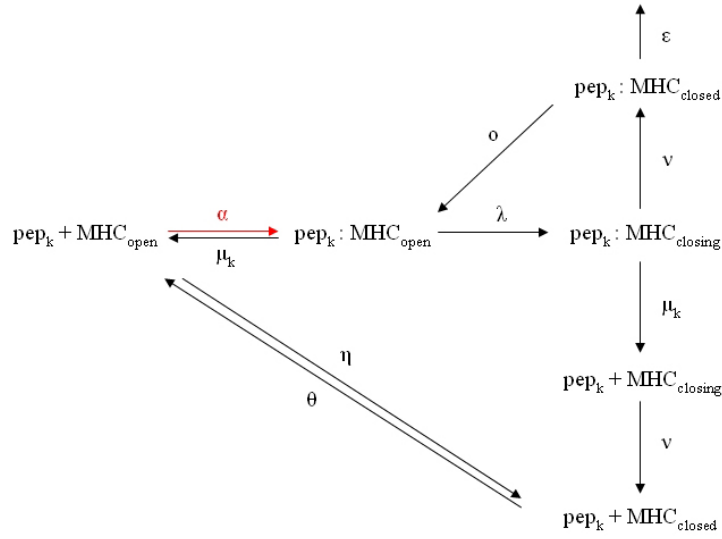


Figure 8: Non-tapasin pathway

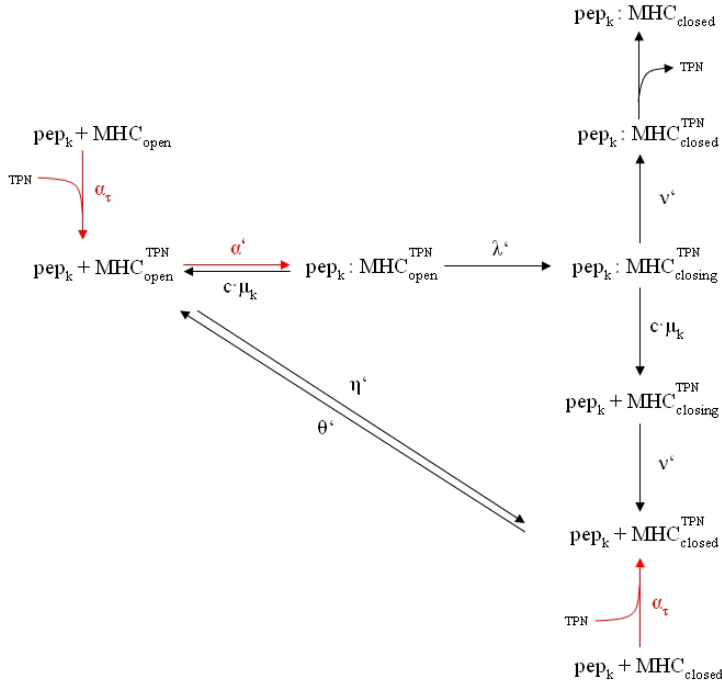


Figure 9: Tapasin pathway

two implementations lies in the code for peptide dissociation. Peptide dissociation is modelled as a single stochastic delay in (a), and as two sequential stochastic delays in (b) (one for the phase of closure initiation and one for the phase of closure). The approach to (a) is intuitive but requires some compli-

MHC_{open}	MHC_{open}
MHC_{closed}	MHC_{closed}
pep	pep
pep: MHC_{open}	MHC_{open_pep}
pep: MHC_{closed}	MHC_{closed_pep}
TPN	TPN
MHC_{open}^{TPN}	MHC_{open_TPN}
MHC_{closed}^{TPN}	MHC_{closed_TPN}
pep: MHC_{open}^{TPN}	$MHC_{open_pep_TPN}$
pep: MHC_{closed}^{TPN}	$MHC_{closed_pep_TPN}$

Table 2: Modelled components and corresponding SPiM processes. $MHC_{closing}$ has no direct correspondence, as it describes an intermediate state coded within other processes.

μ	mu	$c \cdot \mu$	c_TPN*mu
λ	dstart	λ'	dstart_TPN*dstart
ν	dclosed	ν'	dclosed_TPN*dclosed
η	o2c	η'	o2c_TPN*o2c
θ	c2o	θ'	c2o_TPN*c2o
o	dopen		

Table 3: Reaction rate constants and corresponding SPiM parameters

cated coding, while (b) may be counter-intuitive with rather straight-forward code.

(a) If peptide dissociation is modelled as a single stochastic delay, peptide dissociation and molecule closure have to run in parallel, and the outcome of either process influences the outcome of the other. A correct implementation is given below¹. Note that channels `away` and `c` have infinite rate.

```

MHC_open_pep(away:chan,c:chan,open:chan,e:chan)=
  do ?away;MHC_open()
  or delay@dstart;
  delay@dclosed;(do ?away;MHC_closed() or !c;MHC_closed_pep(away,c,o,e))

pep_bound(mu:float,away:chan,c:chan,o:chan,e:chan)=
  do delay@mu;(!away|pep(mu))
  or ?c;pep_captured(mu,away,c,o,e)

```

The bound peptide can either dissociate after a delay at rate `mu`, or it can receive an output `!c` from `MHC_open_pep`, announcing that the MHC complex is closed and the peptide is captured. If it dissociates, it immediately creates a free peptide process `pep` and sends an output `!away` to `MHC_open_pep` in parallel.

The MHC complex can receive an input `?away`, accepting that the peptide has dissociated, or it can start molecule closure after a delay at rate `dstart`. Once closure has been initiated, the complex closes after a delay at rate `dclosed`,

¹The given code refers to the flytrap mechanism in absence of tapasin. Representation channels for the peptide processes are omitted.

but it will inevitably end up in closed conformation. When closure is complete, there are two choices which are uniquely determined by the outcome of peptide dissociation. If the peptide has not dissociated yet, there is no output on channel `away`, and input `?c` in `pep_bound` immediately accepts output `!c`. New processes `MHC_closed_pep` and `pep_captured` are instantiated. If the peptide has dissociated, there is no longer a recipient for output `!c`. Instead, input `?away` immediately accepts output `!away` which has been sent by `pep_bound` upon peptide dissociation. Now a new process `MHC_closed` is created, while `pep` has already been instantiated at the time of unbinding.

A disadvantage of coding peptide dissociation in the peptide process is due to the assumption that dissociation behaviour is affected by the presence of tapasin. For this reason, two separate processes `pep_bound` and `pep_bound_TPN` become necessary.

```
pep_bound(mu:float,away:chan,c:chan,o:chan,e:chan) =
  do delay@mu;(!away|pep(mu))
  or ?c;pep_captured(mu,away,c,o,e)
```

```
pep_bound_TPN(mu:float,away:chan,c:chan,o:chan,e:chan) =
  do delay@c_TPN*mu;(!away|pep(mu))
  or ?c;pep_captured(mu,away,c,o,e)
```

The former is initiated after `pep` binds to `MHC_open` through `a_pep`, while the latter is initiated after binding to `MHC_open_TPN` through `a_pep_TPN`. `pep_bound` dissociates at rate `mu`, `pep_bound_TPN` at rate `c_TPN*mu`. If the peptide does not unbind before molecule closure, either process becomes `pep_captured` and dissociates from tapasin immediately. In the case of molecule reopening, `pep_bound` will be instantiated, regardless of whether the original loading occurred in presence or absence of tapasin. This is correct, as only empty HC/β_2m can bind to tapasin.

(b) Consider the alternative implementation.

```
MHC_open_pep(mu:float,away:chan,c:chan,o:chan,e:chan)=
  do delay@mu;!away;MHC_open()
  or delay@dstart;
  do delay@mu;!away;delay@dclosed;MHC_closed()
  or delay@dclosed;!c;MHC_closed_pep(mu,away,c,o,e)
```

```
pep_bound(mu:float,away:chan,c:chan,o:chan,e:chan) =
  do ?away;pep(mu)
  or ?c;pep_captured(mu,away,c,o,e)
```

Now the flytrap mechanism reduces to a choice between closure initiation and peptide dissociation and, depending on the outcome, a choice between molecule closure and peptide dissociation. The issue of parallel processes is thus resolved, and peptide dissociation and molecule closure can be coded in the same process, here `MHC_open_pep` and `MHC_open_pep_TPN`. Now only one process `pep_bound` is required, and peptide binds to both HC/β_2m and $TPN:HC/\beta_2m$ through the same channel `a_pep`.

Note that, as peptide dissociation is no longer coded in `pep_bound`, the dissociation rate `mu` has to be passed on to `MHC_open_pep`. Also, outputs on channels `away` and `c` are now sent from `MHC_open_pep` and are accepted by

`pep_bound`. The additional channels `o` and `e` correspond to events of MHC reopening and egression.

Finally, note that in the case of peptide dissociation after closure initiation, `delay@dclosed` has to be inserted, or otherwise the closing process would automatically be completed at the time point of peptide dissociation.

The fact that (a) and (b) are formally equivalent follows from the memory-less property of the exponential distribution and is not immediately obvious. To prove equivalence, determine the transition rates for the implied continuous-time Markov chain (Figure 10). The transition rates given in the diagram are obvious from (b).

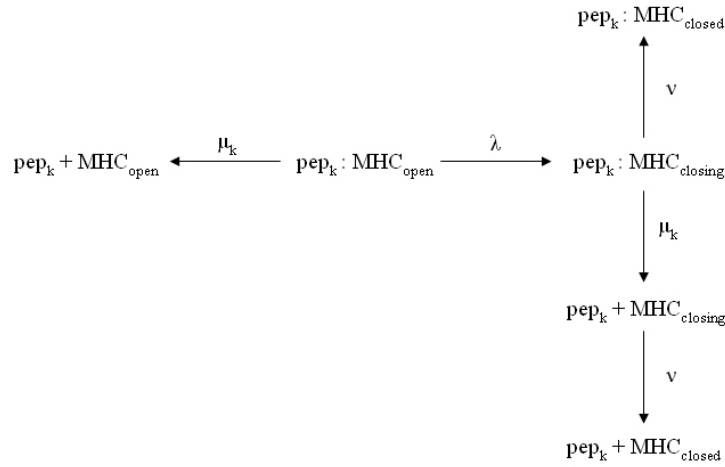


Figure 10: Continuous-time Markov chain as implied in Figure 8

Consider (a). Clearly, transitions ($\text{pep}_k : \text{MHC}_{\text{open}} \longrightarrow \text{pep}_k + \text{MHC}_{\text{open}}$) and ($\text{pep}_k : \text{MHC}_{\text{open}} \longrightarrow \text{pep}_k : \text{MHC}_{\text{closing}}$) occur at rates μ_k and λ respectively. Assume state $\text{pep}_k : \text{MHC}_{\text{closing}}$. Transitions from this state are coded as a race between `delay@dclosed` and `delay@mu`.

```
MHC_open_pep(away:chan,c:chan,open:chan,e:chan)=
  do ?away;MHC_open()
  or delay@dstart;
  delay@dclosed;(do ?away;MHC_closed() or !c;MHC_closed_pep(away,c,o,e))
```

```
pep_bound(mu:float,away:chan,c:chan,o:chan,e:chan)=
  do delay@mu;!away|pep(mu)
  or ?c;pep_captured(mu,away,c,o,e)
```

Suppose `delay@dclosed` wins. Then

```
MHC_open_pep(away:chan,c:chan,open:chan,e:chan)=
  do ?away;MHC_open()
  or delay@dstart;
  delay@dclosed;(do ?away;MHC_closed() or !c;MHC_closed_pep(away,c,o,e))}}
```

```

pep_bound(mu:float,away:chan,c:chan,o:chan,e:chan)=
  do delay@mu;(!away|pep(mu))
  or ?c;pep_captured(mu,away,c,o,e)

```

But channel c has infinite rate. Hence MHC_closed_pep and pep_captured are instantiated after a stochastic delay at rate dclosed , and the transition to $\text{pep}_k : \text{MHC}_{\text{closed}}$ occurs at rate ν . Now suppose delay@mu wins. Then

```

MHC_open_pep(away:chan,c:chan,open:chan,e:chan)=
  do ?away;MHC_open()
  or delay@dstart;
  delay@dclosed;(do ?away;MHC_closed() or !c;MHC_closed_pep(away,c,o,e))

```

```

pep_bound(mu:float,away:chan,c:chan,o:chan,e:chan)=
  do delay@mu;(!away|pep(mu))
  or ?c;pep_captured(mu,away,c,o,e)

```

So pep is instantiated after a stochastic delay at rate μ , and the transition to $\text{pep}_k + \text{MHC}_{\text{closing}}$ occurs at rate μ_k . Only after an additional delay@dclosed , away will fire at infinite rate.

```

MHC_open_pep(away:chan,c:chan,open:chan,e:chan)=
  do ?away;MHC_open()
  or delay@dstart;
  delay@dclosed;(do ?away;MHC_closed() or !c;MHC_closed_pep(away,c,o,e))

```

```

pep_bound(mu:float,away:chan,c:chan,o:chan,e:chan)=
  do delay@mu;(!away|pep(mu))
  or ?c;pep_captured(mu,away,c,o,e)

```

Therefore transition $\text{pep}_k + \text{MHC}_{\text{closing}} \longrightarrow \text{pep}_k + \text{MHC}_{\text{closed}}$ occurs at rate ν . Hence (a) and (b) are equivalent.

3.2.2 Model parameters and discussion

Although peptide binding is now modelled as a two-step process, peptide selection is based on the same principle as before. While the previous model achieved peptide selection by increasing the number of bound and dissociated peptides before the time of egression (or tapasin unbinding), it is now the time until molecule closure that is crucial for selection. As a consequence, peptide selection is largely independent of egression. In fact, it is completely independent of egression in the case of zero MHC reopening rate o as assumed in this section. For $o > 0$, egression does affect the peptide repertoire, as $\text{HC}/\beta_2\text{m}/\text{peptide}$ complexes with high K_a peptide are more stable and thus more likely to egress eventually. This effect is captured by parameters o , μ_k , λ , ν versus ϵ , rather than just μ_k versus ϵ . For sufficiently low o , the effect of egression on the peptide repertoire will thus be limited.

Elliott and Williams argue that tapasin must influence selection of peptides

towards those with low off-rate and allow HC/ β_2 m to bind to peptides that it has previously been unable to effectively acquire [4]. This is consistent with the stated assumption that tapasin has a positive effect on qualitative and quantitative aspects of peptide loading.

In particular, they propose three potential functions of tapasin:

1. Catalysis of the conformational change between the encounter complex and the closed conformation with the effect of an increased dissociation rate of bound peptide. A similar function has been ascribed to HLA-DM, a peptide exchange factor for MHC class II molecules. HLA-DM achieves peptide dissociation rates which are directly proportional to the intrinsic peptide off-rates. The effect is explained by the interruption of hydrogen bonds between the MHC class II molecule and the peptide ligand.
2. Stabilization of the open peptide-receptive conformation. This could have the effect of disrupting key intramolecular hydrogen bonds between HC/ β_2 m and peptide, thus promoting peptide dissociation, as suggested under 1. At the same time, the stabilization could result in a raised energy threshold for the peptide-induced conformational change.
3. Destabilization of the closed peptide non-receptive conformation. In analogy, this would correspond to resetting the flytrap.

1. and 2. correspond to increased peptide dissociation rates ($c > 1$) and/or a decreased closure-trigger rate ($\lambda' < \lambda$). In stochastic π -calculus simulations for both the basic model and the flytrap model, peptide discrimination is observed for accelerated peptide dissociation or prolonged phase of closure initiation (see sections 3.1.3 and 3.2.3).

Destabilizing the closed conformation 3. corresponds to changed rates of spontaneous conformational change in absence of peptide ($\eta' < \eta$, $\theta' > \theta$). A destabilization of the closed conformation as described has no direct effect on peptide selection but does provide an explanation for increased overall peptide loading (see section 3.2.3). Again, this is consistent with the previous model and would provide a satisfactory interpretation of increased peptide binding of HC/ β_2 m in presence of tapasin.

Above, peptide selection was attributed to increased dissociation rates $c > 1$ and/or decreased closure initiation rate $\lambda' < \lambda$. However, in principal, peptide discrimination could also be achieved by slowed molecule closure $\nu' < \nu$. More formally, the conditional distribution for loaded MHC complexes is symmetric in λ , ν :

$$\mathbb{P}(K_a = k) = \left(\frac{(\mu_k + \lambda)(\mu_k + \nu)}{(\mu_{low} + \lambda)(\mu_{low} + \nu)} + \frac{(\mu_k + \lambda)(\mu_k + \nu)}{(\mu_{med} + \lambda)(\mu_{med} + \nu)} + \frac{(\mu_k + \lambda)(\mu_k + \nu)}{(\mu_{high} + \lambda)(\mu_{high} + \nu)} \right)^{-1}$$

Assume peptide selection is achieved through reduced ν . Then peptide loading is more inefficient, as peptide discrimination necessarily involves the dissociation of low K_a peptide, and any peptide dissociation after closure initiation will inevitably lead to molecule closure. The closed HC/ β_2 m complex is no longer peptide receptive and needs to be reopened before it can accept new peptide. The difference between simulation results obtained from either alternative is remarkably small.

Parameter	TPN (1)	TPN (2)	TPN (3)
c_TPN	1.0	100.0	100.0
dstart_TPN	1.0	0.01	0.01
dclosed_TPN	1.0	1.0	1.0
o2c_TPN	0.01	1.0	0.01
c2o_TPN	100.0	1.0	100.0

Table 4: Parameter values for tapasin simulation

There are different ways to interpret the flytrap model. From a structural point of view, an increased activation energy threshold corresponds to decreased $\lambda' < \lambda$. However, molecule closure could be achieved by a protein folding step with several energetically distinct stages, rather than a simple conformational change. Then λ would correspond to the energy threshold for some irreversible change during protein folding, while subsequent changes would be captured by ν . Hence, if tapasin alters the energy thresholds of these subsequent stages, ν would be affected as well.

Conversely, one could think of a simpler rather than more complex process, in which molecule closure is triggered immediately after peptide binding. Then peptide selection is exclusively achieved during the phase of molecule closure, and ν captures any initiation or subsequent stages of the closing process. Such a model would be superior in the sense that it can explain the observed effect of tapasin with fewer free parameters. However, it no longer allows for peptide dissociation without inducing molecule closure.

Thus the two-phase model is the simplest flytrap model which allows for peptide dissociation, peptide capture and failed peptide capture, as suggested in [6]. Nevertheless, it is worth noting firstly that the two-phase flytrap model could be reduced to a one-phase model, and secondly that it provides a correct abstraction for potentially more complex multiple-stage processes underlying "molecule closure".

3.2.3 Simulation results

Table 4 shows three different alternatives for tapasin, combining the functions discussed above. Their effect is illustrated in Figure 11. TPN (1) has no discriminative power but shifts the equilibrium of open and closed conformation towards the peptide receptive form, thus increasing overall peptide loading. TPN (2) has discriminative power but does not affect peptide receptivity of MHC. Interestingly, no peptide discrimination is observed in presence of 100 tapasin molecules, while slightly better peptide selection is achieved for 2000 tapasin molecules, suggesting that the lack of peptide discrimination is at least partly due to tapasin saturation. Finally, TPN (3) possesses both functions and achieves both peptide discrimination and increased overall loading.

Each plot in Figure 11 shows a single representative run of the SPiM simulator. Parameters are set to `dstart=999.0`, `dclosed=1000.0`, `o2c=100.0`, `c2o=0.1` and `epsilon=100.0`.

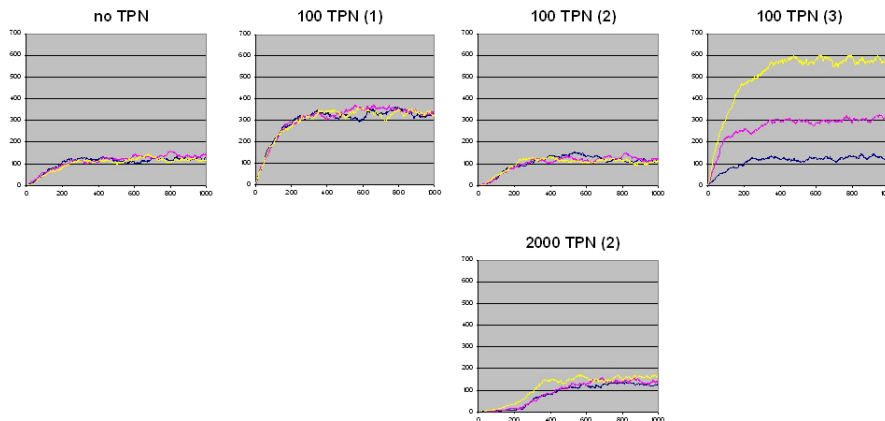


Figure 11: MHC loading in absence and presence of tapasin (MHC parameters $\lambda=999.0$, $\nu=1000.0$, $\eta=100.0$, $\theta=0.1$, $\epsilon=100.0$)

3.3 Time-dependent optimization of peptide repertoire

The previous section explored to what extent the flytrap model allows to vary both peptide selection and the overall level of peptide loading. It was found that the quality of the peptide repertoire is determined by parameters λ , ν and c , which capture the stability of the open HC/ β_2m /peptide complex. On the other hand, the quantity of assembled MHC complexes is influenced by parameters η and θ , which describe the equilibrium of open and closed conformation of HC/ β_2m in absence of peptide, thereby essentially determining the availability of peptide-receptive MHC complexes. So far, MHC reopening was ignored so that the peptide repertoire of egressed complexes was in fact the initially acquired peptide cargo. If the MHC complex is allowed to reopen, the peptide repertoire is optimized over time.

3.3.1 Simulation results

To monitor time-dependent improvement of the peptide repertoire, set `dopen=1.0`. Consider a limited number of MHC complexes by executing 200 processes `MHC_open` at the beginning of the simulation, instead of continuously generating and degrading new processes. Moreover, set `epsilon=0.0`, so that MHC complexes stay in the ER to undergo optimization.

To put some constraints on the parameter space, the following parameters are fixed for all simulations.

`dclosed=1000.0`, following the logic that molecule closure is a relatively fast process once the activation energy threshold is overcome and closing has been initiated.

`o2c=0.0`, `c2o=10.0` ensure that empty MHC complexes are in a peptide-receptive state.

`dstart_TPN=1.0`, `dclosed_TPN=1.0`, `o2c_TPN=1.0`, `c2o_TPN=1.0` This essentially reduces the effect of tapasin to acceleration of peptide dissociation and serves the purpose of constraining the parameter space.

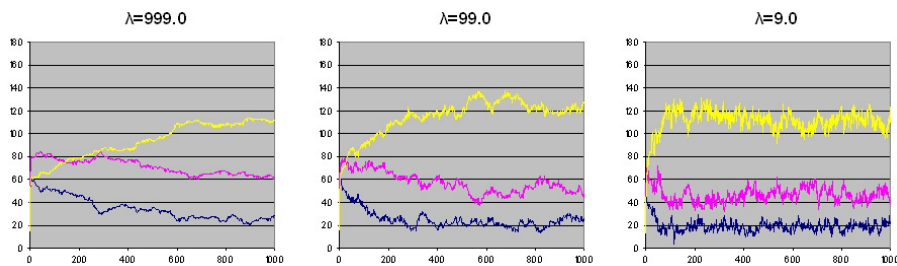


Figure 12: Peptide optimization in absence of tapasin

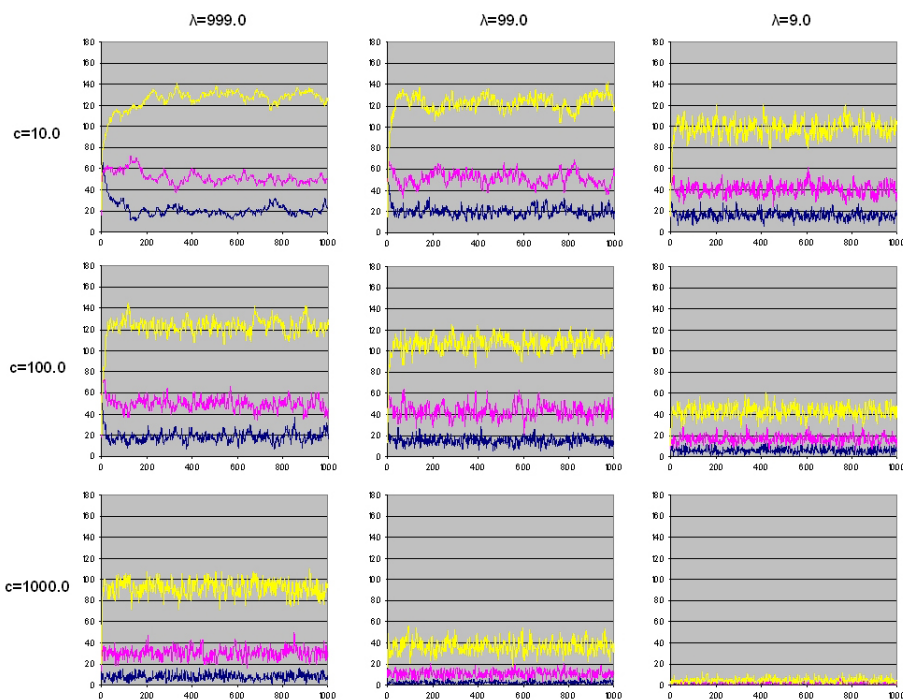


Figure 13: Peptide optimization through tapasin only

Individual plots again present a single representative run of the SPiM simulator. Figure 12 illustrates time-dependent peptide optimization in absence of tapasin, so that peptide loading occurs through the non-tapasin pathway only (Figure 8). Conversely, peptide loading is restricted to the tapasin pathway (Figure 9) in Figure 13. Figure 14 shows optimization in presence of 100 tapasin processes.

The simulation results plotted in Figure 12 demonstrate the effect of reducing λ . While the speed of optimization is improved through higher discriminative power, the equilibrium between open and closed conformation in presence of peptide is shifted towards the open conformation. As previously noted, this reduces the efficiency of peptide loading (the tradeoff between selection and throughput). Moreover, due to MHC reopening, it now also affects the stability

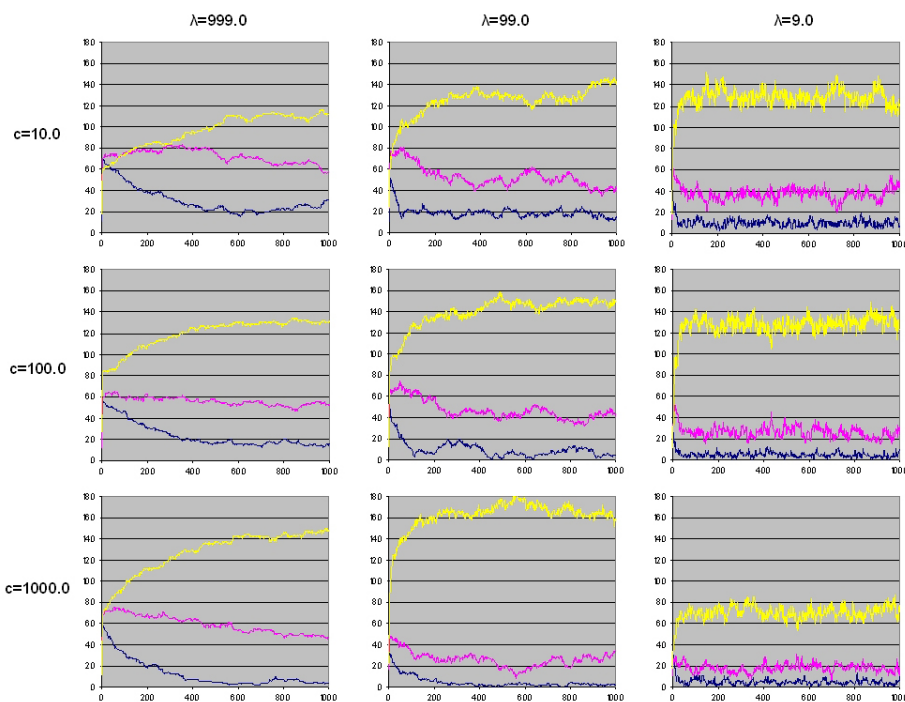


Figure 14: Peptide optimization in presence of tapasin

of HC/ β_2 m/peptide. After MHC reopening, bound peptide has more time to escape before molecule closure. This loss of stability is noticeable in form of greater intrinsic noise. Moreover, for $\lambda = 9.0$, a considerable share of the MHC complexes is either empty or no longer in closed conformation, hence not all 200 MHC processes are plotted in this case. Note that the equilibrium distribution of the peptide repertoire is highly similar for all simulations.

In Figure 13, similar observations can be made. Higher discriminative power, be it through slow closure initiation or fast peptide dissociation, accelerates peptide optimization and simultaneously reduces loading efficiency and stability of the HC/ β_2 m/peptide trimer. Simulation results for $c = 10$ and $c = 100$ (top and middle row) appear to have a similar steady state distribution as observed in absence of tapasin, except for case (100,9). The equilibrium distribution for (1000,999) appears to be slightly more optimal than in the previous cases. For (100,9), (1000,99) and (1000,9), the loaded MHC complex has become too unstable.

Figure 14 shows that the interaction of tapasin and non-tapasin pathways is by no means straight-forward. Remarkably, the steady state distribution of the peptide repertoire is improved considerably in six out of nine cases. Compared to the non-tapasin and tapasin pathway in isolation, more optimal peptide cargo is achieved in all cases except for (10,999), (100,999) and interestingly (1000,9). Except for cases (100,9), (1000,9), MHC stability and loading efficiency matches the non-tapasin pathway and is hardly affected by tapasin.

3.3.2 Discussion

To better understand the limiting behaviour, consider the loading distribution of a generalized flytrap.

For a one-phase flytrap (ignoring closure initiation), have

$$\mathbb{P}_1(K_a = k) = \left(\frac{\mu_k + \nu}{\mu_{low} + \nu} + \frac{\mu_k + \nu}{\mu_{med} + \nu} + \frac{\mu_k + \nu}{\mu_{high} + \nu} \right)^{-1}$$

with limit distribution $D_1 = (0.105, 0.263, 0.632)$ for $\nu \rightarrow 0$. Note that \mathbb{P}_1 is equal to the loading distribution of the basic non-flytrap model with tapasin unbinding-rate τ replaced by closure rate ν_1 .

For the standard two-phase flytrap, have

$$\mathbb{P}_2(K_a = k) = \left(\frac{(\mu_k + \lambda)(\mu_k + \nu)}{(\mu_{low} + \lambda)(\mu_{low} + \nu)} + \frac{(\mu_k + \lambda)(\mu_k + \nu)}{(\mu_{med} + \lambda)(\mu_{med} + \nu)} + \frac{(\mu_k + \lambda)(\mu_k + \nu)}{(\mu_{high} + \lambda)(\mu_{high} + \nu)} \right)^{-1}$$

with limit distribution $D_2 = (0.023, 0.145, 0.832)$ for $\lambda, \nu \rightarrow 0$.

The distributions for loading through tapasin are of course similar, with μ_k , λ , ν replaced by $c \cdot \mu_k$, λ' , ν' . The limiting distribution is now achieved for $\lambda', \nu' \rightarrow 0$ or equivalently $c \rightarrow \infty$.

Under "ideal conditions", the above distributions \mathbb{P}_1 and \mathbb{P}_2 exactly describe the average composition of initially acquired peptide cargo for given loading parameters. This initial composition is improved over time through MHC re-opening and exchange of bound peptide. When the tapasin and non-tapasin pathway are considered separately (Figure 13 and 14), the composition of the peptide repertoire seems to approach a steady state distribution close to D_1 , regardless of the values of λ or c . The lack of further improvement might be due to the fast second closure phase with small potential for peptide selection. Indeed for certain smaller values of ν , the limiting distribution is slightly more optimal than D_1 , but still considerably less so than D_2 (not shown). Loss of stability appears to make more optimal peptide selection impossible.

Remarkably, peptide discrimination shows an extreme increase in presence of tapasin. Loading through tapasin occurs with high discriminative power without affecting the stability of MHC complexes which are currently not bound to tapasin. This trick seems to allow a considerable improvement of all aspects of peptide loading, that is composition of peptide repertoire, loading efficiency and speed of time-dependent optimization.

Interestingly, the described effect of tapasin is also observed for a single-phase flytrap model. Figure 15 was obtained for a simulation run under conditions similar to case (99,1000) in Figure 14, except for the change $\lambda = \infty$, $\nu = 99.0$. This shows that there is no direct correspondence between the phases of the flytrap model and the attained steady state distribution of the peptide repertoire. In particular, the optimal distribution attainable for an n -step flytrap model is not "bounded" by D_n .

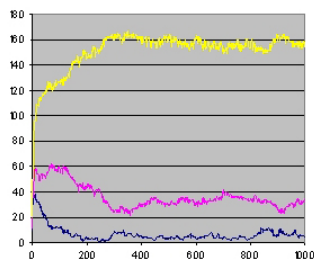


Figure 15: Peptide optimization in presence of tapasin for a single-phase flytrap model ($\lambda = \infty$)

3.4 Model testing

3.4.1 MHC class I heavy chain alleles

Different MHC class I heavy chain alleles exhibit differences in the tapasin-dependent and time-dependent optimization of their peptide repertoire. The best current data on the effect of tapasin with a temporal element is available in [10] (T. Elliott, personal communication). The article by Williams *et al.* describes HLA-B*4402 (B4402), a tapasin-dependent allele, HLA-B*2705 (B2705) a relatively tapasin-independent allele, and HLA-B*4405 (B4405), a tapasin-independent natural polymorphism of B4402.

The results presented in [10] are based on the correlation between the thermostability of a MHC class I complex and the affinity of its peptide cargo. The thermostability of MHC complexes at 4°C, 37°C and 50°C was measured at 30, 60 and 120 minutes after synthesis of the MHC cohort. High thermostability was interpreted as optimally loaded MHC complexes, while low thermostability was interpreted as either suboptimally loaded or empty MHC complexes. The data was obtained for MHC complexes extracted from cells, and not for MHC complexes expressed at the cell-surface only.

In particular, the following observations were made

1. B4402 is incapable of cell surface expression or peptide optimization in absence of tapasin. Moreover, a considerable proportion of MHC complexes is lost during the time course of the experiment. In presence of tapasin, the peptide repertoire undergoes time-dependent optimization, reaching its maximum after two hours. The majority of MHC complexes are maintained.
2. B2705 exhibits cell surface expression in absence and presence of tapasin. In absence of tapasin, the thermostability of MHC complexes improves over time, reaching its maximum after two hours. The increase in thermostability is either due to time-dependent optimization of the peptide repertoire, or loading of previously empty MHC complexes. Only a small proportion of MHC is lost during the time course of the experiment. Tapasin achieves a significant kinetic and qualitative improvement of peptide optimization, with optimization mostly taking place in the first 30-60 minutes. The peptide repertoire obtained in presence of tapasin is less optimal than for B4402.

3. In contrast to B4402, B4405 is capable of cell surface expression in absence of tapasin. It achieves marginally higher thermostability than B4402 in absence of tapasin, but no time-dependent optimization. Tapasin increases thermostability, but the gain is much less substantial than for B4402.

The reported loss of MHC complexes was attributed to dissociation of β_2m following dissociation of low affinity peptides and will hence be ignored.

Thus the three alleles differ in the speed of peptide cargo optimization, the steady state composition of peptide repertoire and the level of cell surface expression.

Time-dependent optimization and equilibrium composition of the peptide repertoire were discussed in 3.3. The third aspect is relevant for B4402, which lacks cell surface expression in absence of tapasin. In the stochastic π -calculus model, no egression translates to absence of `MHC_closed_pep`. This can be achieved by

- (i) free B4402 being present in its closed form only (ν large, θ small), or
- (ii) B4402 being unable to close in absence of tapasin (λ small).

As the experimental data only provides averaged thermostability data, it is impossible to determine whether B4402 complexes in absence of tapasin are actually loaded with peptide or not. Thus (i) is not ruled out a priori, although Williams and colleagues argue in favour of suboptimally loaded MHC complexes, even in absence of tapasin. (ii) assumes that MHC complexes are temporarily loaded with suboptimal peptide, but due to the inability of molecule closure, the complexes are very thermolabile. This hypothesis is at odds with results on a different allele H-2D^b by Springer *et al.* [8], who report the lack of an unstable, fast-dissociating intermediate at early time points in the reaction. Looking at the effect of tapasin, (i) implies a shift of the open-closed equilibrium in absence of peptide towards the open peptide receptive conformation, while (ii) implies lowering the energy threshold for molecule closure. Thus (ii) appears to be inconsistent with the previous interpretation of the function of tapasin, and (i) is assumed in the following section.

3.4.2 Model fitting

It was shown that B4402 and B2705 achieve equal thermostability for exogenously added index peptides [10]. Hence any observed differences in thermostability should be due to differences in the peptide repertoire only, and parameters `dstart`, `dclosed` and `dopen` are set equal for all alleles. Based on the simulation results in 3.3, choose `dstart=999.0`, `dclosed=1000.0`, `dopen=1.0`.

Here parameters `o2c` and `c2o` are qualitatively determined by the cell surface expression. For B4402, the equilibrium between open and closed conformation in absence of peptide is completely shifted towards the non-receptive closed conformation. For B2705 and B4405, it must lie further towards the open conformation. The differences in parameters `o2c` and `c2o` for B2705 and B4405 are less clear and were chosen after consideration of simulation results based on different parameter combinations.

The difference in the effect of tapasin on different alleles is modelled by the extent of tapasin association. This is based on the observation that polymorphisms control the interaction of MHC class I alleles with tapasin and the PLC.

Parameter	Value
c_TPN	100.0
dstart_TPN	0.01
dclosed_TPN	1.0
o2c_TPN	0.001
c2o_TPN	1000.0

Table 5: Tapasin parameters

Parameter		B4402	B2705	B4405
dstart	λ	999.0	999.0	999.0
dclosed	ν	1000.0	1000.0	1000.0
dopen	o	1.0	1.0	1.0
o2c	η	1000.0	10.0	100.0
c2o	θ	0.01	0.1	0.1
a_TPN	α_τ	1.0	0.001	0.00001

Table 6: MHC class I heavy chain parameters

B4402 and B4405 exhibit strong and poor association with the PLC respectively, while B2705 occupies an intermediate position [4].

Despite the allele-dependent interaction with tapasin, it is desirable to model a consistent, allele-independent function of tapasin. Therefore assume a constant multiplicative effect of all tapasin parameters c_TPN , $dstart_TPN$, $dclosed_TPN$, $o2c_TPN$, $c2o_TPN$ (Table 5). Given the suggested dual effect achieved by a stabilization of the open HC/ β_2m /peptide complex, resulting in both an increased energy threshold for conformational change and accelerated peptide dissociation, set $c_TPN=100.0$ and $dstart_TPN=0.01$. Assume no change in molecule closure $ratedclosed_TPN=1.0$. Finally, set $o2c_TPN=0.001$ and $c2o_TPN=1000.0$. This has the effect of tapasin always shifting the equilibrium towards the peptide receptive open conformation. Results are shown in Figure 16. Based on these parameter values, the effect of tapasin can be summarized to stabilization of the open conformation in absence and presence of peptide.

The model has obvious weaknesses. In the case of B4402, it is unclear whether the complete lack of peptide loaded MHC complexes in absence of tapasin is realistic. Moreover, the model is unable to explain the difference in thermostability between B2705 and B4405 in absence of tapasin through the composition of the peptide repertoire alone. Note that in the case of B4405, the open-closed equilibrium is shifted towards the non-receptive closed form, thus reducing the overall number of loaded MHC complexes, similar to B4402 in absence of tapasin. Analyzing the thermostability of all MHC complexes in this case would of course result in reduced average thermostability, as empty MHC complexes are even less stable than suboptimally loaded complexes. For equal peptide receptivity, the composition of the peptide repertoire in absence of tapasin at the end of the simulation would be similar for B2705 and B4405. Finally, the limiting behaviour is not well understood (see 3.3), and it is possible that the true steady state distribution is identical in most cases.

The initial assumptions are possibly too strong. Greater differences in the

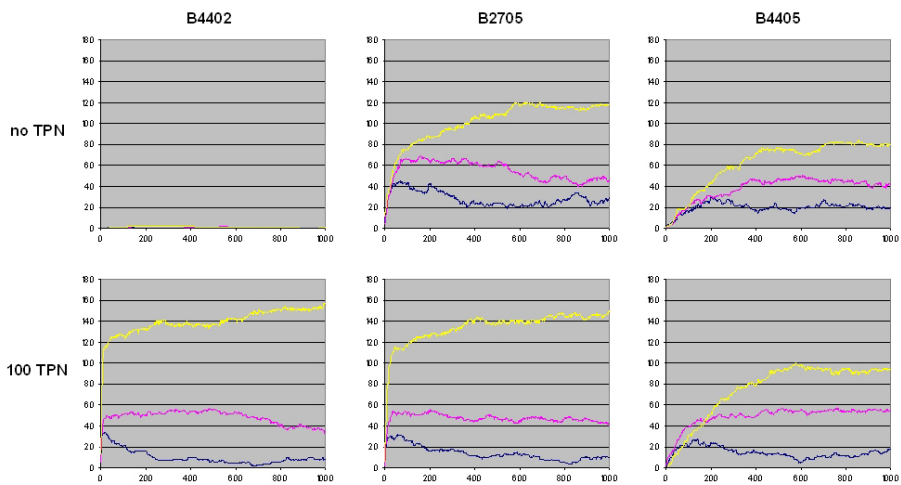


Figure 16: Simulation results

ability to select optimal peptide in absence of tapasin could be achieved for allele-dependent variations in the stability of the open HC/β_2m /peptide complex (determined by λ). Indeed the implications of structural difference can be subtle. For example, the differences in B4402 and B4405 are due to a single amino acid change at position 116. It lies at the opening of the so called F pocket at the end of the peptide binding groove, which contains the C-terminal end of the bound peptide. This region generally appears to be very sensitive to polymorphisms with regard to peptide loading properties. A number of mutations were found to be context-sensitive: A substitution of H to D at residue 114 causes B2705 to lose its ability to self-optimize, even though both B4402 and B4405 have D114. Furthermore, additional changes to B2705 at residues 116 or 152 in the F pocket were shown to recover the self-optimizing ability [4].

4 Conclusion

The large number of cofactors involved in MHC class I assembly make it difficult to determine the exact interaction and function of all components. In particular, the cooperative nature of the peptide loading complex is poorly understood. The presented stochastic π -calculus model focusses on the class I specific accessory molecule tapasin.

A simplified model of peptide loading, assuming no conformational change of the HC/ β_2 m complex, was introduced to provide some intuition for the analysis of the more advanced flytrap model.

Based on the analysis of the advanced model, two main aspects of peptide loading could be attributed to distinct properties of the HC/ β_2 m complex. Peptide selection in favour of peptides with low off-rates seems to be determined by the rate of peptide-induced conformational change, in concert with the rate of peptide dissociation, while the quantitative efficiency of peptide loading can be attributed to the equilibrium between open and closed conformation in absence of peptide. A tradeoff between optimal peptide selection and loading efficiency can be observed. The modelled effects of tapasin are primarily based on tapasin affecting these two variables. For disabled MHC reopening, the peptide discrimination and loading efficiency are interdependent rather trivially, as a peptide discriminating effect of tapasin can only be observed if a considerable proportion of MHC complexes is loaded in presence of tapasin. In the case of MHC reopening, the effect of tapasin can no longer be explained by a simple partitioning of loading pathways, and peptide selection and loading efficiency have a more subtle effect on the composition of the peptide repertoire.

It was possible to confirm the functions of tapasin that had previously been suggested in context of the flytrap concept, in particular a stabilizing effect on the open peptide receptive conformation and a destabilizing effect on the empty closed conformation. The model was fitted to three specific MHC class I heavy chain alleles B4402, B2705 and B4405. However, the suggested parameter values strongly depend on the stated assumptions and do not necessarily provide the best possible fit or the most accurate biological description. Given the large parameter space, assumptions on the underlying biology are indispensable for any fitting or hypothesis testing. For future use, the current two-phase model can be abstracted to a one-phase model to reduce the number of free parameters. Furthermore, the modular approach of stochastic π -calculus holds potential for a more comprehensive model of MHC class I assembly.

5 Acknowledgements

I would like to thank Luca Cardelli and Andrew Phillips at Microsoft Research Cambridge for their dedicated support throughout the project. Also I would like to thank Tim Elliott and Joern Werner at the University of Southampton, who provided valuable insights into the biological aspects.

A SPiM source code

A.1 Basic model

```
(* BASIC MODEL *)

(* directives for sampling and plotting simulation results *)
directive sample 100.0 1000
directive plot
MHC();pep();TPN();MHC_TPN();MHC_pep();MHC_pep_TPN();
!f1 as "pep_low";!f2 as "pep_med"; !f3 as "pep_high";
!b1 as "MHC_pep_low (ER)";!b2 as "MHC_pep_med (ER)";!b3 as "MHC_pep_high (ER)";
!e1 as "MHC_pep_low (cell surface)";!e2 as "MHC_pep_med (cell surface)";!e3 as "MHC_pep_high (cell surface)"

(* reaction channels *)
new a_TPN@1.0:chan(chan)
new a_pep@0.001:chan(float,chan,chan)
new a_pep_TPN@1.0:chan(float,chan,chan)

(* representation channels *)
new f1@0.0:chan new b1@0.0:chan new e1@0.0:chan
new f2@0.0:chan new b2@0.0:chan new e2@0.0:chan
new f3@0.0:chan new b3@0.0:chan new e3@0.0:chan

(* generation *)
val dg_MHC=10.0
val dg_pep=20.0

(* degradation *)
val degrad_MHC=0.1
val degrad_pep=10.0

(* egression *)
val epsilon=1000.0

(* parameters:peptide *)
val mu_low=3.0
val mu_med=1.2
val mu_high=0.5

(* parameters:TPN *)
val c_TPN=100.0
val tau=10.0

let MHC()=
  do ?a_pep(mu,u,e);MHC_pep(mu,u,e)
  or ?a_TPN(u_TPN);MHC_TPN(u_TPN)
  or delay@degrad_MHC

and MHC_pep(mu:float,u:chan,e:chan)=
  do delay@mu;!u;MHC()
  or delay@epsilon;!e

and MHC_TPN(u_TPN:chan)=
  ?a_pep_TPN(mu,u,e);MHC_pep_TPN(mu,u,e,u_TPN)

and MHC_pep_TPN(mu:float,u:chan,e:chan,u_TPN:chan)=
  do delay@mu*c_TPN;!u;MHC_TPN(u_TPN)
  or delay@tau;!u_TPN;MHC_pep(mu,u,e)

let TPN()= (
  new u_TPN:chan
  !a_TPN(u_TPN);(?u_TPN;TPN())
)

let pep(mu:float,r0:chan,r1:chan,r2:chan)= (
  new u:chan new e:chan
  do !r0
  or !a_pep(mu,u,e);pep_bound(mu,u,e,r0,r1,r2)
  or !a_pep_TPN(mu,u,e);pep_bound(mu,u,e,r0,r1,r2)
  or delay@degrad_pep
)
```

```

and pep_bound(mu:float,u:chan,e:chan,r0:chan,r1:chan,r2:chan) =
  do !r1
  or ?u;pep(mu,r0,r1,r2)
  or ?e;pep_egressed(r2)

and pep_egressed(r2:chan) =
  do !r2
  or delay@degrad_MHC

let generate_MHC()=
  delay@dg_MHC;(generate_MHC()|MHC())

let generate_pep()=
  do delay@dg_pep;(generate_pep()|pep(mu_low,f1,b1,e1))
  or delay@dg_pep;(generate_pep()|pep(mu_med,f2,b2,e2))
  or delay@dg_pep;(generate_pep()|pep(mu_high,f3,b3,e3))

run 1 of generate_MHC()
run 1 of generate_pep()
run 100 of TPN()

```

A.2 Flytrap model (a)

```
(* FLYTRAP A *)

(* directives for sampling and plotting simulation results *)
directive sample 1000.0 1000
directive plot
MHC_open();MHC_closed();pep();TPN();
MHC_open_TPN();MHC_closed_TPN();
MHC_open_pep();MHC_open_pep_TPN();MHC_closed_pep();
!f1 as "pep_low"; !f2 as "pep_med"; !f3 as "pep_high";
!b1 as "MHC_open_pep_low";!b2 as "MHC_open_pep_med";!o3 as "MHC_open_pep_high";
!c1 as "MHC_closed_pep_low";!c2 as "MHC_closed_pep_med";!c3 as "MHC_closed_pep_high";
!e1 as "MHC_pep_low (cell surface)";!e2 as "MHC_pep_med (cell surface)";!e3 as "MHC_pep_high (cell surface)"

(* reaction channels *)
new a_TPN@0.0001:chan(chan)
new a_pep@1.0:chan(chan,chan,chan,chan)
new a_pep_TPN@1.0:chan(chan,chan,chan,chan)

(* representation channels *)
new f1@0.0:chan new b1@0.0:chan new c1@0.0:chan new e1@0.0:chan
new f2@0.0:chan new b2@0.0:chan new c2@0.0:chan new e2@0.0:chan
new f3@0.0:chan new b3@0.0:chan new c3@0.0:chan new e3@0.0:chan

(* generation *)
val dg_MHC=10.0
val dg_pep=20.0

(* degradation *)
val degrad_MHC=0.01
val degrad_pep=10.0

(* egression *)
val epsilon=0.0

(* parameters:MHC*)
val dstart=999.0
val dclosed=1000.0
val dopen=1.0
val o2c=10.0
val c2o=1.0
val c_TPN=1000.0

(* parameters:peptide *)
val mu_low=3.0
val mu_med=1.2
val mu_high=0.5

(* parameters:TPN *)
val dstart_TPN=1.0
val dclosed_TPN=1.0
val o2c_TPN=0.001
val c2o_TPN=1000.0

let MHC_open()=
  do ?a_pep(away,c,o,e);MHC_open_pep(away,c,o,e)
  or ?a_TPN(u_TPN);MHC_open_TPN(u_TPN)
  or delay@o2c;MHC_closed()
  or delay@degrad_MHC

and MHC_open_pep(away:chan,c:chan,o:chan,e:chan)=
  do ?away;MHC_open()
  or delay@dstart;
  delay@dclosed;(do ?away;MHC_closed() or !c;MHC_closed_pep(away,c,o,e))

and MHC_open_TPN(u_TPN:chan)=
  do ?a_pep_TPN(away,c,o,e);MHC_open_pep_TPN(away,c,o,e,u_TPN)
  or delay@o2c_TPN*o2c;MHC_closed_TPN(u_TPN)

and MHC_open_pep_TPN(away:chan,c:chan,o:chan,e:chan,u_TPN:chan)=
  do ?away;MHC_open_TPN(u_TPN)
  or delay@dstart_TPN*dstart;
```



```

    delay@dclosed_TPN*dclosed;(do ?away;MHC_closed_TPN(u_TPN) or !c;!u_TPN;MHC_closed_pep(away,c,o,e))

and MHC_closed()=
do ?a_TPN(u_TPN);MHC_closed_TPN(u_TPN)
or delay@c2o;MHC_open()
or delay@degrad_MHC

and MHC_closed_TPN(u_TPN:chan)=
delay@c2o_TPN*c2o;MHC_open_TPN(u_TPN)

and MHC_closed_pep(away:chan,c:chan,o:chan,e:chan)=
do delay@epsilon;!e
or delay@dopen;!o;MHC_open_pep(away,c,o,e)

let TPN()= (
new u_TPN:chan
!a_TPN(u_TPN);(?u_TPN;TPN())
)

let pep(mu:float,r0:chan,r1:chan,r2:chan,r3:chan)= (
new away:chan new c:chan new o:chan new e:chan
do !r0
or !a_pep(away,c,o,e);pep_bound(mu,away,c,o,e,r0,r1,r2,r3)
or !a_pep_TPN(away,c,o,e);pep_bound_TPN(mu,away,c,o,e,r0,r1,r2,r3)
or delay@degrad_pep
)

and pep_bound(mu:float,away:chan,c:chan,o:chan,e:chan,r0:chan,r1:chan,r2:chan,r3:chan) =
do !r1
or delay@mu;(!away|pep(mu,r0,r1,r2,r3))
or ?c;pep_captured(mu,away,c,o,e,r0,r1,r2,r3)

and pep_bound_TPN(mu:float,away:chan,c:chan,o:chan,e:chan,r0:chan,r1:chan,r2:chan,r3:chan) =
do !r1
or delay@c_TPN*mu;(!away|pep(mu,r0,r1,r2,r3))
or ?c;pep_captured(mu,away,c,o,e,r0,r1,r2,r3)

and pep_captured(mu:float,away:chan,c:chan,o:chan,e:chan,r0:chan,r1:chan,r2:chan,r3:chan) =
do !r2
or ?o;pep_bound(mu,away,c,o,e,r0,r1,r2,r3)
or ?e;pep_egressed(r3)

and pep_egressed(r3:chan) =
do !r3
or delay@degrad_MHC

let generate_MHC()=
delay@dg_MHC;(generate_MHC()|MHC_open())

let generate_pep()=
do delay@dg_pep;(generate_pep()|pep(mu_low,f1,b1,c1,e1))
or delay@dg_pep;(generate_pep()|pep(mu_med,f2,b2,c2,e2))
or delay@dg_pep;(generate_pep()|pep(mu_high,f3,b3,c3,e3))

run 1 of generate_MHC()
run 1 of generate_pep()
run 100 of TPN()

```

A.3 Flytrap model (b)

```
(* FLYTRAP B *)

(* directives for sampling and plotting simulation results *)
directive sample 1000.0 1000
directive plot
MHC_open();MHC_closed();pep();TPN();
MHC_open_TPN();MHC_closed_TPN();
MHC_open_pep();MHC_open_pep_TPN();MHC_closed_pep();
!f1 as "pep_low";!f2 as "pep_med";!f3 as "pep_high";
!b1 as "MHC_open_pep_low";!b2 as "MHC_open_pep_med";!b3 as "MHC_open_pep_high";
!c1 as "MHC_closed_pep_low";!c2 as "MHC_closed_pep_med";!c3 as "MHC_closed_pep_high";
!e1 as "MHC_pep_low (cell surface)";!e2 as "MHC_pep_med (cell surface)";!e3 as "MHC_pep_high (cell surface)"

(* reaction channels *)
new a_TPN@1.0:chan(chan)
new a_pep@1.0:chan(float,chan,chan,chan,chan)

(* representation channels *)
new f1@0.0:chan new b1@0.0:chan new c1@0.0:chan new e1@0.0:chan
new f2@0.0:chan new b2@0.0:chan new c2@0.0:chan new e2@0.0:chan
new f3@0.0:chan new b3@0.0:chan new c3@0.0:chan new e3@0.0:chan

(* generation *)
val dg_MHC=10.0
val dg_pep=20.0

(* degradation *)
val degrad_MHC=0.01
val degrad_pep=10.0

(* egression *)
val epsilon=0.0

(* parameters:MHC *)
val dstart=99.0
val dclosed=1000.0
val dopen=1.0
val o2c=0.0
val c2o=10.0
val c_TPN=1000.0

(* parameters:peptide *)
val mu_low=3.0
val mu_med=1.2
val mu_high=0.5

(* parameters:TPN *)
val dstart_TPN=1.0
val dclosed_TPN=1.0
val o2c_TPN=1.0
val c2o_TPN=1.0

let MHC_open()=
  do ?a_pep(mu,away,c,o,e);MHC_open_pep(mu,away,c,o,e)
  or ?a_TPN(u_TPN);MHC_open_TPN(u_TPN)
  or delay@o2c;MHC_closed()
  or delay@degrad_MHC

and MHC_open_pep(mu:float,away:chan,c:chan,o:chan,e:chan)=
  do delay@mu;!away;MHC_open()
  or delay@dstart;
  do delay@mu;!away;delay@dclosed;MHC_closed()
  or delay@dclosed;!c;MHC_closed_pep(mu,away,c,o,e)

and MHC_open_TPN(u_TPN:chan)=
  do ?a_pep(mu,away,c,o,e);MHC_open_pep_TPN(mu,away,c,o,e,u_TPN)
  or delay@o2c_TPN*o2c;MHC_closed_TPN(u_TPN)

and MHC_open_pep_TPN(mu:float,away:chan,c:chan,o:chan,e:chan,u_TPN:chan)=
  do delay@c_TPN*mu;!away;MHC_open_TPN(u_TPN)
  or delay@dstart_TPN*dstart;
```

```

do delay@c_TPN*mu;!away;delay@dclosed_TPN*dclosed;MHC_closed_TPN(u_TPN)
or delay@dclosed_TPN*dclosed;!c;!u_TPN;MHC_closed_pep(mu,away,c,o,e)

and MHC_closed()=
do ?a_TPN(u_TPN);MHC_closed_TPN(u_TPN)
or delay@c2o;MHC_open()
or delay@degrad_MHC

and MHC_closed_TPN(u_TPN:chan)=
delay@c2o_TPN*c2o;MHC_open_TPN(u_TPN)

and MHC_closed_pep(mu:float,away:chan,c:chan,o:chan,e:chan)=
do delay@epsilon;!e
or delay@dopen;!o;MHC_open_pep(mu,away,c,o,e)

let TPN()= (
new u_TPN:chan
!a_TPN(u_TPN);(?u_TPN;TPN())
)

let pep(mu:float,r0:chan,r1:chan,r2:chan,r3:chan)= (
new away:chan new c:chan new o:chan new e:chan
do !r0
or !a_pep(mu,away,c,o,e);pep_bound(mu,away,c,o,e,r0,r1,r2,r3)
or delay@degrad_pep
)

and pep_bound(mu:float,away:chan,c:chan,o:chan,e:chan,r0:chan,r1:chan,r2:chan,r3:chan) =
do !r1
or ?away;pep(mu,r0,r1,r2,r3)
or ?c;pep_captured(mu,away,c,o,e,r0,r1,r2,r3)

and pep_captured(mu:float,away:chan,c:chan,o:chan,e:chan,r0:chan,r1:chan,r2:chan,r3:chan) =
do !r2
or ?o;pep_bound(mu,away,c,o,e,r0,r1,r2,r3)
or ?e;pep_egressed(r3)

and pep_egressed(r3:chan) =
do !r3
or delay@degrad_MHC

let generate_MHC()=
delay@dg_MHC;(generate_MHC()|MHC_open())

let generate_pep()=
do delay@dg_pep;(generate_pep()|pep(mu_low,f1,b1,c1,e1))
or delay@dg_pep;(generate_pep()|pep(mu_med,f2,b2,c2,e2))
or delay@dg_pep;(generate_pep()|pep(mu_high,f3,b3,c3,e3))

run 1 of generate_MHC()
run 1 of generate_pep()
run 100 of TPN()

```

References

- [1] Corrado Priami. Stochastic π -Calculus. *The Computer Journal*, 38(7), 1995.
- [2] Daniel T. Gillespie. Exact Stochastic Simulation of Coupled Chemical Reactions. *The Journal of Physical Chemistry*, 81(25):2340–2361, 1977.
- [3] Andrew Phillips and Luca Cardelli. A correct abstract machine for the stochastic pi-calculus. In *Bioconcur'04*. ENTCS, August 2004.
- [4] Tim Elliott and Anthony Williams. The optimisation of peptide cargo bound to MHC class I molecules by the peptide loading complex. *Immunology Review*, 6, 2005. In press.
- [5] Antony N Antoniou, Simon J Powis, and Tim Elliott. Assembly and export of MHC class I peptide ligands. *Current Opinion in Immunology*, 15:75–81, 2003.
- [6] A. Williams, C. Au Peh, and T. Elliott. The cell biology of MHC class I antigen presentation. *Tissue Antigens*, 59:3–17, 2002.
- [7] Jonathan W. Yewdell, Eric Reits, and Jacques Neefjes. Making sense of mass destruction: quantitating MHC class I antigen presentation. *Nature Reviews Immunology*, 3(12):952–961, 2003.
- [8] Sebastian Springer, Klaus Döring, Jonathan C. A. Skipper, Alain R. M. Townsend, and Vincenzo Cerundolo. Fast Association Rates Suggest a Conformational Change in the MHC Class I Molecule H-2D^b upon Peptide Binding. *Biochemistry*, 37:3001–3012, 1998.
- [9] Andrew Phillips. *The SPiM User Manual*, 2005. Available under www.doc.ic.ac.uk/~anp/spim/Manual.pdf, last visited August 11, 2005.
- [10] AP Williams, CA Peh, AW Purcell, J McCluskey, and T Elliott. Optimization of the MHC Class I Peptide Cargo Is Dependent on Tapasin. *Immunity*, 16(4):509–20, 2002.