

A Simulation for Flavivirus Infection Decoy Responses

James K. Peterson¹, Alison M. Kesson², Nicholas J. C. King³

¹Department of Mathematical and Biological Sciences, Clemson University, Clemson, USA

²Discipline of Paediatrics and Child Health, Sydney Institute for Emerging Infectious Diseases and Biosecurity
Sydney Medical School, The Children's Hospital at Westmead, Westmead, Australia

³Discipline of Pathology, University of Sydney, Sydney, Australia

Email: petersj@clemson.edu

Received 21 January 2015; accepted 9 February 2015; published 12 February 2015

Copyright © 2015 by authors and Scientific Research Publishing Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

Abstract

In this work, we discuss the development of simulation code for a model of the cross-reactive adaptive immune response seen in flavivirus infections. The model specifically addresses flavivirus pathogen virulence in G_0 vs. G_1 cell states. The MHC-I upregulation of resting cells (G_0 state) allows the T-cells generated for flavivirus peptide antigens to attack healthy cells also. The cells in G_1 state are not upregulated as much and so virus hides in them and hence is propagated upon rupture. Hence, this type of model is referred to as a *decoy* model because the immune system is *decoyed* into preferentially recognizing the upregulated cells while the virus actively propagates in another small, but important, cell population. We show that the generic assumption of upregulation via a model which includes the G_0/G_1 differential upregulation leads to immunopathological consequences. We outline the details behind the simulation code decisions and provide some theoretical justification for our model of collateral damage and upregulation.

Keywords

West Nile Virus, Decoy Model, MHC-I Upregulation, G_0 vs G_1 Cell State Upregulation Differences

1. West Nile Virus Infection Models

The family Flaviviridae are single-stranded, plus sense RNA viruses and, with the exception of hepatitis C virus, are all arboviruses, transmitted by mosquitoes and ticks. These viruses cause diseases that include dengue, yellow fever, as well as tick-borne, Japanese and West Nile virus encephalitis. Together these viruses have a

major global impact on human health. They are distributed worldwide with the exception of the polar regions, although a specific flavivirus may be geographically restricted. Notwithstanding, these viruses can expand their geographical distribution, with West Nile virus, for example, emerging in recent years North America, presenting a significant threat to public and animal health. The most serious manifestation of West Nile virus (WNV) infection is fatal encephalitis in humans and horses, as well as in certain domestic and wild birds. The virus is generally maintained in a zoonotic transmission cycle involving mosquitoes and birds. Most individuals will usually experience an in apparent or mild febrile illness, others, a dengue-like illness, while a minority, in particular including the elderly and immunocompromised, may develop encephalitis which can be fatal. The diagnosis is usually made serologically, although the virus may be detected in the blood by molecular techniques, or in tissue culture. In general, humans are regarded as dead-end hosts since they do not generally produce a high enough viremia to enable transmission by mosquitoes. Currently, no vaccine is available and there is no specific antiviral therapy to modify disease outcomes.

Infection of the host by pathogenic viruses begins an interaction that involves evolutionarily co-developed and counter-developed pathways that confer advantage for both organism and host survival. Infection of cells results in the intracellular processing of virus proteins into peptide fragments that bind to nascent class I and II major histocompatibility complex molecules (MHC-I and II) that are displayed on the infected cell surface. This is additional to the normal processing of self-proteins into peptides that occurs in all cells as part of the maintenance of tolerance to self. The combined MHC-virus peptide configuration forms the ligand that enables the recognition of infected cells by virus-specific cytotoxic T lymphocytes (CTL) by the cognate T cell receptor (TcR) expressed on the T cell surface. These CTL cause death of the infected cells by lysis, thereby substantially reducing production of progeny virus. This ultimately leads to eradication of virus from the host, enabling host survival (reviewed in [1]). Flaviviruses such as the neurotropic viruses, West Nile (WNV), Murray Valley encephalitis, Kunjin and Japanese encephalitis viruses, induce an increase in expression of MHC-I and II, as well as other immune recognition molecules, including intracellular adhesion molecule-1 (ICAM-1, CD54), vascular adhesion molecule-1 (VCAM-1, CD106) and E-selectin (CD62E) variously and on a wide variety of cells [2]. WNV-induced increases in cell surface expression of these molecules therefore results in increased efficiency of recognition and killing of infected cells by WNV-specific CTL [3] [4]. Increases in MHC-I concentration on infected cells enable greater numbers of simultaneous TcR-MHC-I-peptide ligand interactions, which enhance the avidity of interaction of virus-specific CTL with the infected target cell. The increased avidity enables the interaction of infected cells with CTL clones that are previously below the recognition threshold by virtue of their low affinity for MHC-virus peptide ligand. Furthermore, a polyclonal anti-viral CTL population may also include low-affinity clones that are self-reactive, *i.e.*, clones that recognize MHC-I-self peptide configurations [4] or even MHC without peptide specificity [5]. Due to the increased avidity of their interaction, these low-affinity, self-reactive clones can now lyse both infected and uninfected target cells that express high cell surface MHC-I concentrations [6] [7]. Furthermore, other non-specific accessory molecules, such as ICAM-1 expressed on the target cell, interacting with its integrin receptor, lymphocyte function-associated antigen-1 (LFA-1; CD11a/CD18), on the T cell, can also increase the avidity of T cell target cell interactions, thus additionally lowering the affinity threshold for T cell recognition and target cell lysis [8]. In addition, interferon- γ (IFN- γ) also strongly increases the expression of MHC and ICAM-1 on potential target cells in the vicinity, thereby increasing their susceptibility to CTL lysis [9].

Because the avidity of interaction between T cells and their targets can be increased via MHC associated (*i.e.*, MHC-I viral peptide and CD8 MHC-I) and non-MHC-associated (*i.e.*, ICAM-1 LFA-1) interactions, any increases in MHC and/or ICAM-1 expression further enhances CTL recognition of target cells. This in turn increases the chances of low-affinity, self-reactive CTL recognizing these target cells. The position of the cell cycle is also important in flavivirus infection. Cells in G0 (resting) when infected with WNV increase their MHC-I expression 6-10-fold, while infected cycling cells (G1, S, G2 + M) increase MHC-I expression by only 2-3-fold [9]. This results in some 10-fold more lysis of infected G0 target cells than cycling cells by the same CTL [3]. Thus, there is enhanced avidity of interaction between CTL and WNV-infected cells in G0, which express relatively high levels of MHC-I, while infected cycling cells that express much lower levels of MHC-I are less easily recognized. Importantly, WNV replicates to a significantly higher titer in cycling cells than in G0 cells. Thus, virus may be eradicated with relatively poor efficiency in a population of infected cycling cells. *In vivo*, since most cells are in G0, we hypothesize that it is the small population of infected cycling cells that

support virus replication while maintaining a low immunological profile, thus increasing the probability of virus transmission [4]. As indicated above, the release of IFN- γ associated with T cell target cell interactions increases MHC and ICAM-1 expression on uninfected cells in the vicinity of virus-infected cells. Thus, high MHC-I expressing (uninfected) targets would be susceptible to lysis by low affinity self-reactive CTL clones. If virus-specific CTL populations included low-affinity self-reactive (*i.e.*, crossreactive) CTL clones, which were able to lyse these uninfected cells, this would result in destruction of uninfected tissue. In the case of an encephalitic virus infection, this collateral damage to uninfected cells would increase the damage to the brain thereby increasing morbidity and mortality. In this way IFN- γ could be responsible for fatal immunopathology in WNV encephalitis in mice [10].

This work builds a model of the host-pathogen interactions for flaviviruses in which we focus on WNV. We build a model as closely tied to the biological literature as possible with a mathematical framework and we show that the model exhibits immunopathology that we argue leads to host death in a small but troubling 3% - 8% of the Flavivirus-infected host population.

All modeling of this nature requires many compromises and many levels of abstraction. We model the infection within a single *abstract* host. We choose initial cell population sizes and immune system components so that our *abstract* version of a host exhibits reasonable responses closely aligned to what we find in the literature. It is not possible to discuss all of the many design decisions that go into such code development, so we only give some of the details here. The reader can go to [11] for the details for the simulation which is written in C++ and Qt.

Simple T Cell Interaction Algorithms

We begin with a modification of a model of T cell recognition first presented by [12] with changes to handle MHC-I upregulation. Receptors on a given T cell recognize a specific foreign protein snippet, a peptide bound to MHC-I we will call ligand c . This ligand binds to its cognate T cell receptor with high affinity. However, all T cell receptors are also exposed to a variety of possible self-peptide-MHC-I ligands which may bind although with measurably weaker affinity. For clarity, let's treat these peptides as a single self peptide d complexed with MHC-I, which has a lower affinity for the virus specific T cell receptor. Hence, in an antiviral immune response, at the T cell recognition level there is always a danger of misrecognition in which the virus specific T cell receptor recognizes and responds to a ligand d instead of c . This leads to the possibility of immune-mediated destruction of uninfected cells in the host, *i.e.* immunopathology. Following [12], we note the recognition process in T cell receptors includes several steps. Let R denote the T cell receptor. Then the foreign peptide MHC-I ligand c and R collide at rate k_{on} to form a complex $[cR]$ in which the ligand is bound to the receptor. This receptor-ligand complex also disassociates at rate k_{off} thereby unbinding the ligand c from the receptor R . This leads to the simple dynamics

$$\frac{d[cR]}{dt} = k_{on}[c][R] - k_{off}[cR] \quad \text{where } [\] \text{ is the usual notation for concentration of a substance. At steady state,}$$

$$\text{we then have } [cR]' = 0 \text{ leading to } [cR] = \frac{1}{k_c}[c][R] \text{ where } k_c = \frac{k_{off}}{k_{on}}.$$

When the ligand binds to the T cell receptor, it triggers a signal transduction pathway inside the T cell which leads to activation of the T cell. Also, after ligand binding, the receptor undergoes a series of modifications such as phosphorylation at several sites. Let's assume the T cell receptor transduction pathway has five phosphorylation sites. The complex is now renamed $cR-QQQQQ$ where Q denotes a site not presently phosphorylated. Phosphorylation can then create complexes of the form $cR-PQQQQ$ (one site is phosphorylated), $cR-PQQQQ$ (two sites) and so forth to $cR-PPPPP$ (all sites phosphorylated). Each of these modifications are reversible to the original complex cR . Transduction of the T cell signal is not triggered until all the modifications have been made. Hence, this introduces a *delay* between the ligand binding and the T cell signal generation. This suggests only the ligands that stay bound to the T cell receptor long enough to it to become fully phosphorylated can activate the T cell.

From our discussions above, we see the probability that cR remains bound for a time t after binding is related to the *off* rate. Hence, letting p_c denote this probability, we have $p_c(t) = e^{-k_{off}t}$. Signaling only occurs after a delay of τ seconds after the complex forms. Therefore, the probability per ligand that the T cell is

activated is the same as the probability the ligand stays bound longer than τ seconds. Thus, the rate of T cell activation in the presence of a concentration $[c]$ of the viral peptide MHC-I T cell receptor complex $[cR]$ is $A_{\text{virus}} = e^{-k_{\text{off}}\tau}$. And the self peptide d gives rise to a rate of T cell activation $A_{\text{self}} = e^{-k'_{\text{off}}\tau}$, where we assume τ is the same for both. The viral ligand c dissociates from the receptor at a slower rate than the self ligand d due to its stronger bond with the receptor. Thus, $k_{\text{off}} < k'_{\text{off}}$. We see the viral ligand spends more time bound to the receptor than the self one. The error rate for recognition is thus $F = \frac{A_{\text{self}}}{A_{\text{virus}}} = e^{-(k'_{\text{off}} - k_{\text{off}})\tau}$. The literature

suggests a typical $k_{\text{off}} = 1 \text{ sec}^{-1}$ and $k'_{\text{off}} = 10 \text{ sec}^{-1}$ with $\tau = 1.5 \text{ sec}$. Hence, $F \approx e^{-(10-1)1.5} \approx 10^{-6}$. Therefore, long delays can enhance the fidelity of the recognition process. As Alon states, this enhancement comes at a cost: a long signal transduction delay implies that more of the viral ligands can dissociate from the T cell receptor before T cell signaling is triggered, leading to a loss of sensitivity. So loss of sensitivity is acceptable because of the improved specificity, *i.e.* the discrimination between viral and self ligand recognition.

Now, add the WNV to this model of recognition. What changes with upregulation of the number of MHC-I peptide ligands on the surface of the infected cells? As parameters of affinity of the ligand-receptor interaction, we assume that k_{off} , k'_{off} and τ are intrinsic and not altered by the the MHC-I upregulation caused by the WNV infection. As production of anti-viral interferon by infected cells will also increase MHC-I expression on uninfected cells in the local vicinity, we will assume that if the MHC-I peptide ligands are upregulated by a factor M , then the new error rate is scaled by the same factor; *i.e.*, if the upregulation factor is $u = M$, we have $F_{\text{upreg}} = \frac{A_{\text{self}}}{A_{\text{viral}}} = Me^{-(k'_{\text{off}} - k_{\text{off}})\tau}$. Hence, if upregulation is approximately 6 - 10 fold, we see

$F_{\text{upreg}} \approx 6 \times 10^{-6} \approx 10^{-5}$ which allows for a significant recognition of self ligands and consequent signal transduction leading to an increase in collateral uninfected cell damage. In other words, M is a measure of the avidity of the T cell-target cell interaction, which is a function of MHC-I peptide ligands expressed on the cell surface. We therefore have theoretical reasons to suspect that WNV infections can lead to healthy cell loss.

2. One Host Flavivirus Models

The model is based on a theoretical *decoy* model of [13], which simulates the cellular adaptive immune response to flavivirus *in vivo*. In ([13] and [14]), it was argued that natural killer cells did not aid in the recovery of a host from a flavivirus infection. Hence, to investigate this interaction most simply, we choose to model only an adaptive immune response. We now describe the model in detail and justifying the assumptions necessary for its implementation.

The main variables we use are listed in **Table 1(a)**. The cells labeled **FIC0** are infected resting cells G_0 and those labeled **FIC1** are infected and actively dividing, G_1 . The virus level y_0 is a measure of viral load in real data in plaque-forming units (pfu) per cell but its exact definition is not important to the simulation. We initialize the healthy cell populations in our simulations to $(216)^3 \approx 10$ million healthy cells. We could initialize for larger or smaller populations but this is concrete enough to make our points. It is not possible to keep track of detailed information on such a large number of healthy cells individually. Instead, we model our simulation universe as a cube which is 216 on a side. As is common in many simulations, we use a number of auxiliary class objects. In the simulation, addresses of each cell range from 0 to approximately 10 million. These integer addresses correspond to addresses in a 3D box of size $216 \times 216 \times 216$. This box is then divided into cubes with an edge size of 27 which gives 512 cubes. Each cell is located in one of these cubes b , where $0 \leq b \leq 511$. Hence, we have several different cell addressing choices.

1) A cell has an address a which is an integer from 0 to $216^3 - 1$. This address can also be expressed as a triple (x, y, z) which corresponds to its location in 3D space.

2) A cell lives in a cube whose address is (c_x, c_y, c_z) with c_x , c_y and c_z in the range 0 to 7. These addresses are determined by the rewriting (x, y, z) as $x = c_x \times (27) + r_x$, $y = c_y \times (27) + r_y$ and $z = c_z \times (27) + r_z$ where r_x , r_y and r_z are the integer remainders.

3) Inside the cube, the remainders (r_x, r_y, r_z) give the the local cell address (r_x, r_y, r_z) .

Table 1. Variables and probabilities. (a) Variables, (b) Probabilities.

(a)

Variable	Description
y_0	Virus level
y_1	Uninfected cells
y_2	FIC0 : infected resting cells
y_3	FIC1 : infected dividing cells
y_4	T cell population
y_5	New infections population
y_6	New FIC0 cells
y_7	New FIC1 cells
y_8	New virus from FIC0 lysis
y_9	New virus from FIC1 lysis
y_{10}	New recognized FIC0 , FIC1
y_{12}	FIC0 upregulated at time step
y_{13}	FIC1 upregulated at time step
y_{14}	Uninfected cells killed at time step

(b)

Probabilities	Description	Value
p_0	prob. virus infects uninfected cells	$1.0e-3$
p_1	frac. resting infected cells frac. FIC0 cell	0.99
p_2	fraction dividing infected cells frac a FIC1 cell, $1-p_1$	0.01
p_3	prob. immune system kills free antigen	0.001

The basic rates and probabilities we use are listed in **Table 1(b)** and **Table 2(a)**. The nominal values do change with different simulation needs, but they are listed to give a base from which we will diverge as necessary. New viruses are added to the virus population due to lysis of **FIC0** and **FIC1** cells. In our model, p_9 changes at each iteration and the other probabilities are set randomly around the base values given in **Table 1(b)**. There are also some additional parameters as shown in the bottom panel of the table. The use of these parameters, rates and probabilities are discussed in the following sections.

2.1. Infected Cell Modeling

For dividing infected cells, our model also uses IFN- γ and ICAM-1 upregulation. Upon recognition of their cognate ligand, MHC-I-peptide, by the T cell receptor (TcR), cytotoxic T cells secrete IFN- γ which upregulates the cell surface concentration of MHC-I peptide ligand on the infected cell surface. Thus, MHC-I-peptide upregulation is induced by both the WNV infection and IFN- γ . As an accessory adhesion molecule in T cell-target cell interactions, upregulation of ICAM-1 further increases the avidity of the interaction between the TcR and the cognate MHC-I-peptide-ligand. ICAM-1 is also upregulated by IFN- γ . When an infected cell is recognized and killed by a T cell, the IFN- γ secreted by the T cell upregulates the cell surface concentration of both MHC-I and ICAM-1 on all infected and uninfected cells in a neighborhood of the recognized killed cell. We keep track of this changing upregulation level for the infected cell with the variable U_γ .

We could handle the neighborhood upregulation of MHC-I simply. Each upregulated cell belongs to a coarse

Table 2. More probabilities and parameters and a sample upregulation strategy. (a) More probabilities and parameters; (b) A sample upregulation strategy.

(a)		
Parameters	Description	Value
p_4	frac. coarse cube upreg.	0.001
p_5	growth factor uninfected cells	$1.0e-9$
p_6	resource limit uninfected cells	$(216)^3$
p_7	number coarse cell upreg.	20
p_8	T Cell clonal expansion mult.	0.03
p_9	FIC1 viral growth mult. p_9 faster than in FIC0	10.0
p_{10}	free antigen base per cell lysis	1000.0
p_{11}	free antigen standard dev. per cell lysis	100.0
p_{13}	standard dev. for p_0 is $0.1 \times p_0$	$0.1p_0$
p_{14}	collateral damage base	500
p_{15}	collateral damage standard dev.	0.05
p_{16}	self peptide pool size 2^p	$p = 14$
p_{17}	peptides self frac.	0.5
p_{18}	upreg. factor FIC0 cells	9
p_{19}	max. number peptides displayed	40

(b)		
Resting		
Peptide	Type	Integer range
0	WNV	from $\{0, \dots, 8191\}$
\vdots	\vdots	\vdots
5	WNV	from $\{0, \dots, 8191\}$
6	Self	from $\{8192, \dots, 16383\}$
\vdots	\vdots	\vdots
23	Self	from $\{8192, \dots, 16383\}$
Dividing		
Peptide	Type	Integer range
0	WNV	from $\{0, \dots, 8191\}$
1	Self	from $\{8192, \dots, 16383\}$
2	Self	from $\{8192, \dots, 16383\}$
3	Self	from $\{8192, \dots, 16383\}$

cell (c_x, c_y, c_z) . We could then upregulate all infected cells that belong to this coarse cell. The difference between the number of infected cells found in a coarse cell and the total number in the coarse cell, 19683, then gives us an estimate of how many uninfected cells have also been upregulated. This upregulation of cell surface MHC-I and ICAM-I concentration increases the probability that an infected cell will be recognized and killed. Unrecognized infected cells continue to move towards a lytic event which releases progeny viruses into the coarse cell. We model the expression of MHC-I-peptide ligands on the surface of an infected cell as follows. Living cells display MHC-I molecules that bind short peptides within a groove. Together these form a ligand recognized by the TcR. These peptides are derived from both self and non-self proteins.

As a modeling strategy, we make the assumption that there are $2^{13} = 8192$ peptides that are viral non-self peptides. The model can use any 2^N of course, but it is easier to see the thread of our argument with a specific power. This number has been chosen because it is a power of 2 and hence has some advantages in simulations and it was large enough to be pertinent. The self-peptide pool in reality is very large. We will let the self-peptide

pool be of the same size as the non-self pool, *i.e.*, 8192. To model peptides, we will number them from 0 to $2^{14} = 16383$ (in general 2^{N+1} , and divide them in half: viral peptides are labeled from 0 to 8191 and self, from 8192 to 16383. We know that a cell has many MHC-I-peptide ligands on its surface, but we do not want to model all of them. We assume that in a given unit cell surface area, there are 4 MHC-I-peptide ligands recognizable by the TcR, if not upregulated, and a multiple of these if there is upregulation of the MHC-I molecule on the cell surface. Thus, if the upregulation factor is 6, then the upregulated cell would have 24 ligands. In a WNV-infected cell without upregulation, we will assume that the 4 ligands per unit area presented on the cell surface consist of 3 self peptides and 1 WNV peptide. Then, if an infected cell is upregulated by a factor of N , the cell would present $3N$ self peptides and N WNV peptides. For example if the upregulation factor is 6, we would see 18 self peptides and 6 WNV peptides per unit area on the infected cell. The non-upregulated (G_1) cells are handled in a similar fashion except that instead of 24 displayed ligands each time step, we only use 4. One of them is considered viral and the other three are self. Hence, for an upregulation factor of 6, the vector V in D has the form shown in Table 2(b) for resting infected cells and dividing G_1 infected cells. We would have similar data for upregulation factors other than 6, of course.

We create new **FIC0** cells as needed by calling the method **SetFIC0()** which is discussed in [11]. When we create a new **FIC0** cell, we assign it to a randomly chosen coarse cube. We create this coarse cube address by randomly choosing integers u , v and w in the range 0 to 215 (see the cell address discussion earlier). We maintain a list of **FIC0** and **FIC1** cells throughout our simulations and each such cell has the data vector D associated with it, which has the components shown in Table 3(a).

Here a and b are as previously described. The variable t_L is a counter which is increased for each time unit the infected cell is alive. Once t_L exceeds the allowable life time of a typical infected cell, the cell is removed from the simulation. The vector V contains $4N$ integers where N is the amount of upregulation the infected cell has had. For example, if the upregulation level is 6, then the vector would be size 24. The integers refer to the proteins in our simulation which are expressed on the surface of the cell and are therefore available for recognition, as described earlier. Finally, the variables e and U_γ are the infected cell's *avidity* and upregulation state, respectively. We will discuss how these variables are updated next. Our coding decisions and implementation strategies and choices for these matters are discussed fully in [11].

Table 3. Components of D and the coefficient of variation, $C(\Theta)$. (a) Components of D ; (b) The coefficient of variation.

(a)

V	$\{V_0, \cdots, V_{4N}\}$
e	0.0
t_L	0
U_γ	1.0
a	$0 \leq a \leq 10,077,695$
b	$0 \leq b \leq 511$

(b)

Upregulation θ	$r(\Theta)$	$C(\Theta)$
0	4.5	0.3925
1	3.5	0.4450
2	2.5	0.5266
3	1.5	0.6798
4	1.0	0.8326
5	0.7	0.9951
6	0.5	1.1774
7	0.3	1.5200
8	0.1	2.6328
9	0.05	3.7233

2.2. Infected Data Updates

Our model uses both IFN- γ and ICAM-1 upregulation. IFN- γ is secreted by cytotoxic T cells upon recognition of their cognate ligand; this upregulates both MHC-I peptide ligand and ICAM-1 on the cell surface. MHC-I and ICAM-1 upregulation are also induced by WNV infection. We discuss the reasoning behind our model of the upregulation issue later in Section 4. This section explains our strategy for IFN- γ upregulation as well as the killing of uninfected cells that are susceptible to lysis by low affinity anti-viral T cells as argued by the decoy hypothesis [13]. We will refer to such uninfected cell deaths as **collateral damage**.

At each time point in the simulation, new infected cell's are created and added to the **FIC0** or **FIC1** list. For these newly created infected cells, we initialize their age and recognition value to 0; and set their initial IFN- γ value to 1.0. Hence, as the simulation runs, there is a pool of previously created infected cells and infected cells newly created at this time. As soon as an infected cell is created, we calculate its avidity for the T cells which will interact with it at this time. Once the avidity is calculated, we multiply it by the current IFN- γ value of the cell (recall it is 1.0 at this time). Once a cell is infected, we assume there is a time lag before there is ICAM-I upregulation due to the infection. This time delay is set by the variable **ICAMThreshold** in the simulation code. Currently, this is set to be 48 time steps and since a time step corresponds to 15 minutes, we see ICAM-I is not upregulated until after 12 hours. This is the **only** time in the simulation that ICAM-I upregulation is used.

Hence, at each time in the simulation, we have lists of infected cells whose avidities are known. Infected cells start with their intrinsic avidity which is determined by their displayed MHC-I complexes and the particular mix of T cells that are present to interact with them. Each of these cells also has a life time which is updated each simulation step. However, after these computations are done, we then determine how many of these infected cells are to be recognized and lysed or who have lived long enough to rupture. Hence, there are two events: one is the infected cell recognition which initiates IFN- γ upregulation which enhances any infected cells avidity which can receive the IFN- γ signal released by the recognized cell and two: the infected cell death event which does not release IFN- γ but does release progeny virus. We keep track of the coarse addresses of the cells targeted for recognition and removal and we increase the avidity of these cells by the IFN- γ multiplier. We do this up to a maximum multiplier amount which is set by the simulation. Then, at each subsequent simulation step, we check to see if this *enhanced avidity* exceeds the value needed for recognition. Thus, at each time step, the infected cells removed are any cells whose avidity (newly created or enhanced) allows recognition. Hence, the value of an infected cell's recognition value increases over the life of the simulation as upregulation of ligand and ICAM-1 spreads through the neighborhood cells.

We need to look at the details of how we compute the avidity of an infected cell. However, first we will discuss T Cell modeling.

3. T Cell Initialization Code

In this simulation, we create a population of T Cells which can kill WNV-infected cells and if the decoy model is correct, uninfected cells as well. The function we use to do this in is **Create T Cells ()**. The target cell population is divided into big cubes for convenience. At each time step in the simulation, new target cells are infected. New **FIC0** (infected, not dividing) and **FIC1** (infected, dividing) cells are created.

At the time they are created, we calculate the avidity of the T cell-target cell interaction according to the following model. In the simulation, there are 5 West Nile Virus (WNV) peptides that can be placed in 5 MHC-I grooves. The T Cells that can kill an infected cell are then modeled using a distribution of TcR affinities. Each of the T Cells has its highest affinity for one of the 5 MHC-I-WNV peptide ligands. The affinity calculation for each TcR-ligand interaction is essentially a normal distribution. There are 21 points in the bell curve with the one at the center corresponding to the maximum affinity for given MHC-I-WNV peptide ligand. That leaves 10 to the right of the center and 10 to the left each with progressively decreasing affinity. This number 10 is a simulation parameter. These 20 other affinity values then must correspond to how the TcR interacts with a cell via the MHC-I-peptide ligand. To each of these 20 points, we associate self peptides (the ones numbered above 8192). At the start of each simulation, we pick these $20 \times 5 = 100$ peptides randomly. This gives 5 TCRs which have the potential to interact with 20 MHC-I-self peptide ligands (lower affinity than the center one) and the one WNV peptide. At each time step, we choose to compute the **FIC0** or **FIC1** affinity/avidity using a fraction of the full T Cell capability. Thus, instead of calculating the affinity using all $5 \times (20 + 1) = 105$ affinity calculations for each self peptide in the pool, at each time step, we choose a fraction

(now 45%), of these, approximately 47. Thus, we randomly pick 47 numbers from 0 to 104. For example, if one of these numbers is 65, we think of 63 as $3 \times 21 + 2$ and so the number 65 corresponds to the WNV-peptide 3 and the self-peptide 2. The number 77 would then be $3 \times 21 + 14$ corresponding to WNV peptide 3 and self peptide 14. So this collection of 47 numbers gives us a mix of T Cells (TcR) corresponding to the 5 WNV-peptides we selected as well as the other numbers which correspond to the self-peptides.

We assume the avidity of a T Cell to bind to a WNV-infected cell via MHC-I-peptide ligand depends on the amount of MHC-I and other surface molecule upregulation. If we label this upregulation factor as θ , we assign the affinity of peptide i using a normal distribution to be $a_i = e^{-r(\theta)i^2}$. The parameter $r(\theta)$ determines the coefficient of variation of this bell curve. The coefficient of variation, $c(\theta)$, is the width across the bell curve corresponding to an ordinate value of 0.5; this is given by the formula $c(\theta) = \sqrt{\ln(2)/r(\theta)}$. We want the curve representing the affinity of T Cells which recognize MHC-I-peptide ligand to be very narrow with no upregulation of ligand on the target cells ($\theta = 0$) and to widen as θ increases up to our maximum value of $\theta = 9$. We achieve this by choosing the $r(\theta)$ values as shown in Table 3(b). The third column in the table shows how the coefficient of variation of the curves varies with the value of θ . This behavior is illustrated in Figure 1 where the circles on each of these plots represent the values from $i = 0$ (the affinity of the TcR for MHC-I-WNV-peptide) to $i = \pm 10$, showing the decrease in affinity as it moves away from the center where there is the maximum of 1. In the actual code, we adjust these values so that they scatter randomly around the center value and we also use $i + \varepsilon_i$ where ε_i is a random perturbation of i . Note that for small values of θ , the generated affinities are quite small for $i \geq 1$. We can easily adjust this affinity model to make the affinity curves much narrower. We can therefore adjust the affinities we use in our avidity calculations so that they depend on the upregulation factor. The simulation would typically generate $r(\theta)$ such as these for an MHC-I upregulation factor of 9: WNV-Peptide T Cell 0 would have $r(9) = 0.0478649$; WNV-Peptide T Cell 1, $r(9) = 0.0515279$ and so forth. Note all of these are slight perturbations from the base value of 0.05 shown in Table 3(b) which is the base $r(9)$ for a MHC-I upregulation of $\theta = 9$. Then, we compute the actual affinities we will assign to each self-peptide.

Infected Cell Avidity Calculations

Now let's look at a typical infected cell of class FIC0. If the WNV MHC mediated upregulation is 9 fold, in our model, this FIC0 cell has MHC-I grooves that hold populations of peptides that are comprised proportionately from 27 self peptides and 9 WNV peptides. These peptides are chosen randomly from the WNV and self pool, as before. Now at each time point, we use 47 of the T Cells created according to the scheme discussed above. It seems reasonable to model in this way as we get a random selection of a large population of

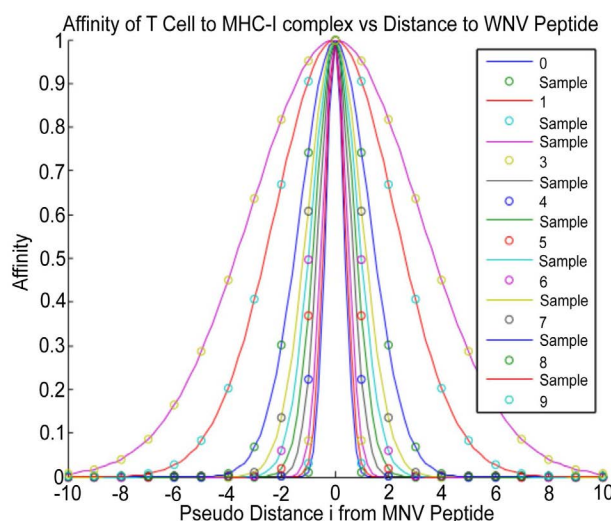


Figure 1. The T Cell affinities vary with the amount of MHC-I upregulation.

T Cells with a range of affinities interacting at each time point. For example, at a given time point, the 47 T Cells we pick could be as follows: the first WNV peptide is 25 and associated with it are the self peptides {9005,10707,14805,8634}. We have now selected 4 of the 47 T Cells. The second WNV peptide is 379 and we select 9 of its self peptides. We have thus selected 14 of the 47 possibilities. Then the third WNV peptide is 3568 and for it we select 13 of its self peptides giving a total of 27 so far. We finish by selecting WNV peptides 4911 and 8 of its self peptides giving 35 total and WNV peptide 7802 and 12 of its self peptides to finish out the 47. In a sense, we have selected 47 peptides to match with the peptides on the surface of the **FIC0** cell. We can either think of this as 47 separate T Cells or as 5 T Cells (one for each WNV peptide) with a range of affinities. The T Cells are *assembled* from these randomly selected numbers. When these numbers are randomly chosen, we do not allow for the same number to be chosen twice. The chances of having the same WNV peptide in a T Cell are small and so we have set up the simulation model to make sure we can not get matches in the random selection process.

We can therefore think of these T Cells as having a response to peptides other than the WNV peptide; we can assemble a T Cell with highest affinity for WNV peptide whose recognition can also include as many as 20 self peptides. Thus, the model we use can create as many T Cells as there are ways to take 47 peptides out of a set of 105. For example, if the 47 numbers randomly created included 65, 69, 77 and 81, this would correspond to a T Cell whose WNV peptide is the third one and whose associated self peptides are the ones stored as self peptide 2, 6, 14 and 18. That T Cell would then be used in the affinity computations at that time step.

To calculate the avidity of these 47 T Cell interactions with this **FIC0** cell, we find out which of the 27 self and 9 WNV peptides expressed on the surface of this **FIC0** cell are recognized. We then sum over all of the affinities that result from the matches. If this sum is 1.0 or more, this **FIC0** cell will be targeted for lysis at the next time step. This recognition will then trigger the IFN- γ upregulation and the ICAM-1 upregulation on the remaining target cells. This is done simply. The original avidity is multiplied by appropriate multipliers (say 1.5 for IFN- γ and 1.2 for ICAM-1 upregulation, respectively) to obtain the new avidity. We do this calculation for all the infected **FIC0** cells, maintaining them in a list. We add them when they are created and delete them upon T Cell recognition and lysis or upon removal because they have exceeded their lifetime limit.

Our process begins by noting out how many active T Cells and infected cells there already are. Then we find the number of new **FIC0** cells that have been created, which indicates where to start the affinity calculations. The previous **FIC0** cells in the list have already had their affinity computed. We are now at the place in the list where we need to calculate the affinity for our newly infected cell. To calculate the affinity, we look at the peptides associated with the infected cell and, for the infected cell, we also store their lifetime and IFN- γ upregulation factor. We have already chosen 47 random T Cells. Each of these corresponds to an integer k which can be written as $k = i \times (21) + j$. Each active T Cell, T_k , therefore has an associated peptide W_{ij} and affinity A_{ij} where i is the WNV peptide number and j is the number of the self-peptide. To indicate the dependence on the number k , we will also add a superscript k . Hence, the active T Cell, T_k , has an associated peptide W_{ij}^k and affinity A_{ij}^k . Then, for each peptide, V_p , on the surface of the infected cell, we check if it matches a peptide that can be recognized by the TcR and add its associated affinity. Let $\Theta(V, W)$ denote the check $\Theta(V, W) = 1$ if V is the same peptide as W and $\Theta(V, W) = 0$, otherwise. If we let Φ denote the infected cells avidity, we then have $\Phi = \sum_{p=0}^4 \sum_{i=0}^{46} \Theta(V_p, W_{i,j}^k) A_{ij}^k$. The 47 active T Cells at this time step are a mix of the 5 WNV peptides, so the value of Φ could be larger than 1 indicating a recognition event. Then, we multiply Φ by the IFN- γ multiplier factor, T_γ making sure we don't exceed the maximum IFN- γ upregulation, T_γ^{\max} , allowed: $\Phi \rightarrow \min(\Phi T_\gamma, T_\gamma^{\max})$. We also allow for ICAM-1 upregulation if the cell has been infected long enough. If the current life of the infected cell exceeds the ICAM threshold time, T_I , we reset Φ again by multiplying by the ICAM-1 upregulation factor, F_I . Of course, if this value exceeds the maximum ICAM-1 upregulation amount, F_I^{\max} , we set Φ to be T_I^{\max} : $\Phi \rightarrow \min(\Phi F_I, F_I^{\max})$.

4. Handling Collateral Damage

To find the avidity of the uninfected cells in each cube (the approximately 20,000 or so), we randomly assign

the self peptides for an uninfected cell and using the full 105 T Cell affinity possibilities, we compute the avidity. To do this, we assume the T Cells are spread uniformly through the cubes and since there are so many uninfected cells compared to infected in general, it would make sense to think all these T Cells could interact with the uninfected cells. When we model an infection, at each time step in a given simulation, we need to estimate collateral damage. We do this by developing an estimate of how much collateral damage occurs during an infection assuming a certain level of upregulation. This estimate is done before we perform any infection simulations. We build these estimates for the upregulation levels from 1 to 9 at least 3 times. The estimates are then averaged over the number of collateral damage computations we have done to provide the collateral damage amounts we use in any of our infection simulations. A typical collateral damage calculation is done as follows. In each cube, we

- choose the self peptides for each of these $\sim 20,000$ cells randomly from the self peptide pool.
- for these $\sim 20,000$ cells, compute their avidities for the 5 T Cells with all their 20 self peptides. We will assume that each uninfected cell expresses a certain number of self peptides on its surface. This number is arbitrary, but clearly, if the entire universe of self peptides were on the cell, then all uninfected cells would be recognized and potentially lysed. We will assume each uninfected cell expresses 3 or 4 randomly chosen self peptides on their surface which is a small fraction from the self-peptide universe of 8192 choices. Our rationale for this is as follows. In any cell, peptide snippets are taken from all self-proteins, processed by molecular machinery, placed in MHC-1 cradles and the resulting complex is transported to the cell's surface. There are a huge number of self-proteins and a limited number of MHC-1 molecules. Hence, at any given moment, a cell displays some fraction of the possible universe of self-peptides as well as a portion of the possible foreign peptides. In addition, MHC-1 molecules are constantly being recycled into component parts for reuse and new MHC-1 molecules with new self-peptides installed in their antigen presenting grooves are being generated. Thus, we can assume only a fraction of the possible self-peptides are displayed at any time and the members of this fraction are probably constantly changing. The immune system generated T Cells that can recognize self-proteins then interact with this fraction of displayed self-peptides generating an affinity. In a typical human, the self-peptide universe is probably on the order of 10^{12} and $\sim 0.05\%$ of this is 5×10^8 displayed self-peptides, which is a reasonable approximation to what we would find in a typical cell. Since our universe size is 8192, this amounts to displaying the fraction $\sim 0.05\%$ of the self-peptide universe; approximately 4 of the self-peptides. In our affinity calculations here, therefore we assume each uninfected cell expresses 3 or 4 randomly chosen self peptides on their surface.

We also know only a fraction of potential T Cells interact with any given uninfected cell at a particular time. Hence, choosing a small number of self peptides to be expressed on each of our uninfected cells and then allowing our T Cell interaction to determine avidity is essentially a more quantitative version of a simple probabilistic law. Such a law might have the form of assuming a certain percentage of uninfected cells are recognized at each time step. However, it is then harder to handle IFN- γ mediated upregulation. Also here, we don't use the 45% fraction chosen randomly at each time step. Instead we do the affinity calculation using the full range of T Cells. This gives us avidities for each cell in the cube; *i.e.* $\sim 20,000$ avidities.

Note it is not clear how we should model these avidities. If we simply sum the affinities, it is quite possible that we would have avidities that exceed 4 or more for a given cell. If the killing threshold is set to 1, this could generate an excessive number of uninfected cell deaths. We will divide our summed affinities by 5 to make it harder to reach the killing threshold. In essence, this is a statistical approach to the calculated avidity. In our calculations, we choose this last approach for our calculated avidities. Our models assume 5 WNV peptides and hence, we divide all the computed avidities by 5 to generate the avidity of each cell in the cube.

- assume when a **FIC0** or **FIC1** cell is killed, there is IFN- γ mediated upregulation of these 20,000 cells to some degree. We have the maximum IFN- γ upregulation set 2^0 , so if the IFN- γ multiplier is set to 1.5, then after 7 upregulations, 1.5^7 is approximately 17. Hence, after 7 upregulations, the upregulation approaches its maximum value and upregulation of these cells stops. Since we are using the full universe of available T Cells, if an uninfected cell reaches maximum IFN- γ mediated upregulation without being lysed, this cell is no longer able to be destroyed by the immune system's response. The cell will then live out its normal life and be removed following normal cellular metabolic rules. Hence, maximally upregulated cells die by natural causes since the computed avidity for these cells implies they have not been recognized by any of the T Cells.

- next, go through the 20,000 avidities just calculated and count how many have a high enough avidity to be killed as they are. Call this number M_0 because there is no upregulation yet. Then, go through the remaining avidities just calculated and count how many have a high enough avidity to be killed with their avidities multiplied by 1.5. Subtract the previous number M_0 and call this number M_1 . This gives the number of additional cells killed after 1 upregulation. We then continue this process: go through the remaining avidities and count how many have a high enough avidity to be killed with their avidities multiplied by 1.5×1.5 . Subtract the previous 2 numbers M_0 and M_1 and call this number M_2 . This gives the number of additional cells killed after 2 upregulations. We keep doing this until you have performed 7 upregulations which approximates maximal upregulation. When complete, for each cube we now have the numbers M_0, M_1 through M_7 .
- now in any infection simulation, we model uninfected cell collateral damage in the following way. At each time step in an infection simulation, we know **FIC0** and **FIC1** cells are killed and their recognition by T Cells generates IFN- γ mediated upregulation signals to the cube they belong to. It takes several time steps, but eventually, enough upregulation signals occur within a given cube so that $M_0 + \dots + M_7$ cells are being killed each time step. This gives us a way to count collateral damage somewhat quantitatively. Note we are assuming, consistent with the acute phase of infection, the upregulation signal is a one way action: there is no mechanism by which this can be down regulated in this model.

The Coarse Update Model

We will assume that our simulation model maintains its level of uninfected cells indefinitely in the absence of infection using a standard logistics growth model. Recall y_1 denotes the uninfected cell level in our model. Hence, we assume the uninfected cells follow a logistics growth law: $y_1'(t) = \alpha y_1(t)(L - y_1(t))$ where the initial condition $y_1(0)$ is less than the resource allocation L . For us, we want to maintain the level $L = 216^3 \approx 10 \times 10^6$. This means the uninfected cells grow at the rate αL , $(y_1')_{\text{growth}} = \alpha L y_1$, with decay proportional to the square of their population $(y_1')_{\text{decay}} = -\alpha y_1^2$. If the growth rate of the cell population is on the order of 1%, we would expect $\alpha L = \alpha (216)^3 = 0.01$. Hence, a good choice for α is $\alpha = 9.92 \times 10^{-10} \approx 10^{-9}$. We set these values in the simulation as $\alpha = p_5$ and $L = p_6$. Note, in discrete form the logistics growth law gives $y_1(t+1) = y_1(t) + \alpha y_1(t)(L - y_1(t))$. Even if the infection has reduced the uninfected cell population to a fraction of the initial level, say 3×10^6 , the update we see at the next time is

$$y_1(t+1) \approx 3 \times 10^6 + 10^{-9} (3 \times 10^6) (10^7 - 3 \times 10^6) = 3 \times 10^6 + 21,000.$$

Hence, about 21,000 cells are created at this time step even though the population is very far below the steady state level of 10^7 . The typical biological response to such a depressed cell level would be relatively small and we want the simulation parameters to be chosen to preserve that level of response.

We estimate collateral damage as follows:

- y_5 denotes the number of new infections. In the simulation, we keep track of the number of new **FIC0** removals (n_{ri}^0) and the number of new **FIC1** removals (n_{ri}^1) in the i^{th} cube. We then compute the fraction $a_i = (n_{ri}^0 + n_{ri}^1) / y_5$ where y_5 is the current total number of infected cells. The value of a_i is set to 1 if this calculation exceeds 1. This fraction allows us to estimate what the fraction of cells in a cube are likely to be infected at this time step.
- at each time step, we keep track of the number of upregulation signals sent to each coarse cube and store this value in the variable S_i where i is the coarse cube number. Letting N denote the number of cells in a coarse cube and we see the fraction S_i / N estimates how much upregulation (and hence collateral damage) is being done in a given coarse cube.
- we have stored the average $\sum M_i$ values for each cube for each possible upregulation value. Denote these values by Σ_i^Θ where Θ is the upregulation level and i is the index of the coarse cube. We randomly perturb the Σ_i^Θ values around their base values to allow for variation. If we denote this perturbation by ϵ_i we have the updated value $(1 + \epsilon_i) \Sigma_i^\Theta$.

- we now have the number

$$c_i = \frac{n_{ri}^0 + n_{ri}^1}{N} \times \frac{S_i}{N} \times (1 + \epsilon_i) \Sigma_i^\Theta.$$

This estimates the collateral damage in the coarse cube i .

- we allow for the possibility that there can be a multiplier effect. The number Σ_i^Θ is the collateral damage estimate for each cell in the cube. Each cell in a cube which is infected sends out upregulation signals which effect additional uninfected cells. This multiplier is the parameter p_7 . For example, if $p_7 = 8$, we allow for an eight fold multiplier effect. Hence, the full collateral damage estimate for cube i is

$$C_i = p_7 \times \frac{n_{ri}^0 + n_{ri}^1}{N} \times \frac{S_i}{N} \times (1 + \epsilon_i) \Sigma_i^\Theta.$$

- we then sum over all cubes to get the full collateral damage estimate, $y_{14} = \sum C_i$.

5. The Dynamics of the Model

To analyze the dynamics of the model we are building, we will need some auxiliary variables and constants which are listed in [Table 4\(a\)](#) and [Table 4\(b\)](#). In [Table 4\(b\)](#), the variable name **IFN_gamma_neighborhood** is replaced by **IFN_gam_nbhd** and **ActiveTcellPercentage** becomes **ActiveTcellPer** to make the table less wide.

There are also a number of nonlinear functions which compute various things about the variables and their interactions. These functions are listed below with their mathematical function names as well as the names they have been given in the code we use for the simulation.

$\Phi_0 \equiv \text{SetFIC0}(y)$: The variable y denotes the new **FIC0** cells. Let the number of these new cells be N_0 . This function picks random peptides to populate the MHC complexes of N_0 new **FIC0** cells created at this current time step.

$\Phi_1 \equiv \text{SetFIC1}(z)$: The variable z denotes the new **FIC1** cells. Let the number of these new cells be N_1 . This function picks random peptides to populate the MHC complexes of N_1 new **FIC1** cells created at this current time step.

$\mathcal{R} \equiv \text{ChooseRandomSubset}(M, N)$: This chooses M unique elements from a list of N things (*i.e.* there are no repeats).

$\Psi_0 \equiv \text{CalculateFIC0Avidities}(z)$: This computes the affinities for **FIC0** cells given an **FIC0** population z .

$\Psi_1 \equiv \text{CalculateFIC1Avidities}(w)$: This computes the affinities for **FIC1** cells given an **FIC1** population w .

$\mathcal{Z}_0 \equiv \text{UpdateL}_0(f)$: We update the lifetimes of the cells in the **FIC0** list f .

$\mathcal{Z}_1 \equiv \text{UpdateL}_1(g)$: We update the lifetimes of the cells in the **FIC1** list g .

$\Omega_0 \equiv \text{RemoveFIC0Cells}(z)$: Here, we determine which **FIC0** cells in the list z have gone past their lifetime and so lyse as well as how many are recognized.

$\Omega_1 \equiv \text{RemoveFIC1Cells}(w)$: Here, we determine which **FIC1** cells in the list w have gone past their lifetime and so lyse as well as how many are recognized.

$\Gamma_0 \equiv \text{UpregulateIFN}_0(f)$: This function IFN- γ upregulates all infected cells in a certain neighborhood of a recognized cell given the current list of cells, f , to upregulate. It returns an estimate of the number of healthy cells so upregulated.

$\Gamma_1 \equiv \text{UpregulateIFN}_1(g)$: This function IFN- γ upregulates all infected cells in a certain neighborhood of a recognized cell given the current list of cells, g , to upregulate. It returns an estimate of the number of healthy cells so upregulated.

The Time Evolution of the Model

We now discuss the simulation dynamics. In what follows, we will use subscripts $t+1$ to denote the values of variables at the next time step and t to indicate the value at the current time step. We show the update equations in the table below which are calculated in the order shown. The simulation first must initialize all the T cells we use.

Table 4. Auxiliary and additional variables. (a) Auxiliary, (b) Additional variables.

(a)

Variable	Meaning
F_0	FIC0 list
F_1	FIC1 list
A	active T cell sites list
\mathcal{A}_0	FIC0 T cell affinity
\mathcal{A}_1	FIC1 T cell affinity
L_0	FIC0 lifetime
L_1	FIC1 lifetime
R_0	FIC0 lysis
R_1	FIC1 lysis
K_0	FIC0 T cell mediated lysis
K_1	FIC1 T cell mediated lysis
U_0	IFN- γ upreg. FIC0 cell list
U	IFN- γ upreg. FIC1 cell list
F_γ	IFN- γ upreg. factor
F_γ^{\max}	maximum IFN- γ upreg.
T_I	FIC0 ICAM-1 upreg. time
F_I	ICAM upreg. factor
F_I^{\max}	Maximum ICAM-1 upreg.

(b)

Name	Meaning	Symbol
numprots	Peptide-MHC ligands on infected cell surface.	N_p
keysize	Number Upreg. Coarse Cubes for infected cell.	N_R
ActiveTcellPer	fraction of T cell clones able to recognize ligand	γ
WNV_size	num. flavivirus peptides in T cell config.	N_{WNV}
ActiveTcell_size	number γN_{WNV} .	N_{TA}
virals	num. viral peptides. $N_V = p_{17} 2^{n_{16}}$.	N_V
recog_percent	ratio $\frac{N_{TA}}{N_V}$.	r_p
IFN_gam_nbhd	frac. uninfected cells in IFN- γ upreg. neighborhood recog.	f_{IFN}
HC_recog_prob	product $f_{IFN} \gamma r_p$.	p_{HC}
HCFraction	$(y_{10})/(y_6 + y_7)$ if denom. not zero, else 0.1	

- Free virus, y_0 , is removed by the immune system with a certain probability, p_3 and virus is added to the pool by the lysis of **FIC0** and **FIC1** cells. We assume that **FIC1** cell lysis contributes p_9 times as much as the lysis of **FIC0** cells. From experimental data, $p_9 \approx 10$, so if N free antigen was contributed by the **FIC0** lysis, then $10N$ would be added by the lysis of **FIC1**. Thus, the amount of free virus is modeled by **new free virus** = **old free virus** + **(virus released due to FIC0 cell lysis)** + **(virus released due to FIC1 cell lysis)** - **(virus removed by immune system prior to infection)**. This is modeled mathematically by $y_0(t+1) = y_0(t) + y_8(t) + p_9 y_9(t) - p_3 y_0(t)$ which is Equation (1).

$$y_0(t+1) = y_0(t) + y_8(t) + p_9 y_9(t) - p_3 y_0(t), \text{ New freeantigen} \quad (1)$$

- Uninfected Cell infections, y_5 , are handled next. We set **new uninfected cell HC infections** = **(Probability of uninfected cell infection)** \times **old amount of virus present**, or $y_5(t+1) = p_0 y_0(t)$, Equation (2).
- Infected cells which are not dividing, are the **FIC0** cells, y_6 . We model their dynamics as **new FIC0 cells at the next time step** = **a percentage of new infections**, $y_6(t+1) = p_1 y_5(t+1)$, Equation (3).

$$y_5(t+1) = p_0 y_0(t), \text{ New cellinfections} \quad (2)$$

$$y_6(t+1) = p_1 y_5(t+1), \text{ Non dividing infected cells, FIC0} \quad (3)$$

- Infected cells which are dividing are the **FIC1** type, y_7 . Their dynamics are modeled as **new FIC1 cells at next time step** = **percentage of new infections** or in mathematical terms $y_7(t+1) = p_2 y_5(t+1)$ which is Equation (4).
- The number of **FIC0** cells, y_2 , is given by **total FIC0 cells at next time step** = **old FIC0 cells** + **new FIC0 cells**, $y_2(t+1) = y_2(t) + y_6(t+1)$, which is Equation (5).

$$y_7(t+1) = p_2 y_5(t+1), \text{ Dividing infected cells, FIC1} \quad (4)$$

$$y_2(t+1) = y_2(t) + y_6(t+1), \text{ Total FIC0 cells} \quad (5)$$

- The number of **FIC1** cells, y_3 , at the next time is then **total FIC1 cells at next time step** = **old FIC1 cells** + **new FIC1 cells** or more quantitatively, $y_3(t+1) = y_3(t) + y_7(t+1)$, Equation (6).

$$y_3(t+1) = y_3(t) + y_7(t+1), \text{ FIC1 cells} \quad (6)$$

For newly created cells of type **FIC0** and **FIC1**, we assign the MHC-I-peptide ligands that are expressed on their surface by a call to the functions **SetFIC0**, Φ_0 , and **SetFIC1**, Φ_1 . In the code, this is done via the function calls **SetFIC0(y)** and **SetFIC1(y)**. The functions Φ_0 and Φ_1 are used to determine randomly the proteins expressed on the surface of an infected cell as discussed in Section 2.1. As long as there are new **FIC1** cells, $y_7 > 0$, we build the peptides bound to the MHC-I molecules on the surface. These are infected G_1 cells and so they are less likely to be recognized and lysed by T cells, since the MHC-I is upregulated to a lesser extent on these cells. Hence, the call to Φ_0 builds $y_6(t+1)$ data vectors V of the appropriate size for **FIC0** cells. We express this mathematically as $F_0(t+1) = \Phi_0(y_6(t+1))$ (Equation (7)).

$$F_0(t+1) = \Phi_0(y_6(t+1)), \text{ Update FIC0 list} \quad (7)$$

The call to Φ_1 builds $y_7(t+1)$ vectors V of size appropriate for **FIC1** cells, $F_1(t+1) = \Phi_1(y_7(t+1))$, which is listed as Equation (8). The new vectors above are then added to the current list of **FIC0** and **FIC1** cells as data. Then, we choose a random subset of the WNV-specific T cells to recognize and lyse the infected cells. Recall at each time step we assume there is a heterogeneous population of T cells which can recognize a fraction of the 105 possible peptide ligands. We designate these as *active*. In our simulation, we let 45% be *active* at each time step. We store the peptides that can be recognized in the **ActiveTcellsIndex** and use these peptides to compute affinities. The function \mathcal{R} chooses N_{TA} indices out of the possible list $\{0, 1, \dots, N_{WNV} - 1\}$ to use as the peptides for which the T cells generated at this time step have an affinity, $A(t+1)$. This can be expressed quantitatively as $A(t+1) = \mathcal{R}(N_{TA}, N_{WNV})$ which is Equation (9).

$$F_1(t+1) = \Phi_1(y_7(t+1)), \text{ Update FIC1 list} \quad (8)$$

$$A(t+1) = \mathcal{R}(N_{TA}, N_{WNV}), \text{ Active T Cell components} \quad (9)$$

We then calculate affinities for the **FIC0** and **FIC1** lists. In the simulations, these lists can easily reach 800,000 in size, so traversal of these data structures has a serious impact on computational time. So we need to determine the avidity of each new active **FIC0** cell clone for each new **FIC0** cell. We do this with the **Calculate FIC0 Avidities** function, labeled Ψ_0 . Previous avidities are already stored in our **FIC0** data structures. We add the new avidities to the list with the call **Calculate FIC0 Avidities (y)**. Any of the newly infected cells whose avidity exceeds the threshold value that triggers T cell destruction is counted in the variable **new_FIC0_kills**. This is stated mathematically as $\mathcal{A}_0(t+1) = \Psi_0(y_6(t+1))$ which is Equation (10). We then do the same sort of thing for new **FIC1** cells where the **Calculate FIC1 Avidities** function is labeled Ψ_1 . In the call to **Calculate FIC1 Avidities**, we also count the number of newly infected cells that are killed and store this value in the variable **new_FIC1_kills**. The resulting equation has the mathematical form

$\mathcal{A}_1(t+1) = \Psi_1(y_7(t+1))$ Equation (11). In these equations the argument $y_6(t+1)$ is the value of the new **FIC0** cells and $y_7(t+1)$ is new value of the **FIC1** cells.

$$\mathcal{A}_0(t+1) = \Psi_0(y_6(t+1)), \text{ Update FIC0 avidities} \quad (10)$$

$$\mathcal{A}_1(t+1) = \Psi_1(y_7(t+1)), \text{ Update FIC1 avidities} \quad (11)$$

The avidity calculations are implemented as previously discussed in Section 3.1. Infected cells whose avidity values enable recognition will then generate IFN- γ which will mediate MHC-I upregulation. We first remove the **FIC0** cells and then the **FIC1** cells. To remove the **FIC0** cells, we

- get the current size of the **FIC0** list of cells.
- update the lifetime values of each cell in this list.
- we remove **FIC0** cells whose lifetime has been exceeded.
- we remove **FIC0** cells whose avidity values trigger T Cell destruction.
- Since y_2 is the number of **FIC0** cells, each removal and kill, decreases y_2 .
- cells dying in the absence of T cell recognition release their viral contents and increase the value of y_0 . The amount of free antigen so generated is determined by a random perturbation around the base value p_{10} multiplied by the number of dying cells. The resulting amount of free antigen is stored in y_8 .
- On the other hand, the antigen in the cells destroyed by T Cells is lost and does not increase the y_0 value. However, each destroyed cell has an associated coarse cube address which we keep track of in a list. Some of the cells in this cube will undergo IFN- γ mediated upregulation. We keep track of the number of T Cell mediated lytic events in this time step in y_{10} .

Essentially, we repeat these steps when we remove **FIC1** cells. Thus, the **FIC0** and **FIC1** removal algorithm first sets the elapsed time on all infected cells and an infected cell will die automatically if its life time is exceeded. This is just a matter of incrementing the life value t_L in each cell in $F_0(t+1)$ and $F_1(t+1)$. This amounts to a traversal through each list. The simple functions that do this are called $\mathcal{Z}_0(F_0(t+1))$ which returns $L_0(t+1)$, the updated list of **FIC0** lifetime values. This is written mathematically as $L_0(t+1) = \mathcal{Z}_0(F_0(t+1))$ and is shown as Equation (12). The similar equation, $L_1(t+1) = \mathcal{Z}_1(F_1(t+1))$ for the updated list of **FIC1** lifetime values is listed as Equation (13) where $\mathcal{U}_1(F_1(t+1))$ is the function which does the lifetime updating for the **FIC1** list.

$$L_0(t+1) = \mathcal{Z}_0(F_0(t+1)), \text{ Update FIC0 lifetimes} \quad (12)$$

$$L_1(t+1) = \mathcal{Z}_1(F_1(t+1)), \text{ Update FIC0 lifetimes} \quad (13)$$

As part of this list traversal, recognized infected cells are lysed by a call to the functions $\Omega_0(F_0(t+1))$ (this computes y_8 , resets y_2 and sets the y_{10} portion due to **FIC0** cells) and $\Omega_0(F_1(t+1))$ (this computes y_9 , resets y_3 and adds the y_{10} due to **FIC1** cells). We let $R_0(t+1)$ denote the number of **FIC0** cells which have exceeded their lifetimes and hence, undergo lysis without T Cell recognition and add released antigen to y_0 . We let $K_0(t+1)$ be the number of **FIC0** cells which are recognized by the immune system and killed

(Equation D14).

$$(R_0(t+1), K_0(t+1), U_0(t+1)) = \Omega_0(F_0(t+1)), \text{ Remove FIC0 cells} \quad (14)$$

Similarly, the number of FIC1 cells exceeding their lifetimes or recognized by the immune system and killed are denoted by $R_1(t+1)$ and $K_1(t+1)$, respectively (Equation D16).

$$(R_1(t+1), K_1(t+1), U_1(t+1)) = \Omega_1(F_1(t+1)), \text{ Remove FIC1 cell} \quad (15)$$

We then need to upregulate IFN- γ in a selection of addresses around each FIC0 cell lysed by a T cell. The address a in the data D for each **FIC0** cell that has been lysed by a T cell belongs to a corresponding coarse cell in the immune model, (c_x, c_y, c_z) . We handle the IFN- γ in the method **Set Neighborhoods** as follows:

- the cube addresses associated with destroyed **FIC0** and **FIC1** cells are stored in a vector which we convert into a list.
- the list of cube addresses which contain a destroyed infected cell is sorted and duplicates are then removed. This gives us the size of the removed list without duplicates, N_R .
- the **FIC0** and **FIC1** lists of infected cells are examined and a mapping is set up between the address of the infected cells data and the cube address the infected cell reside in.
- the number of **FIC0** cells that are upregulated at this time step is stored in y_{12} ; similarly, the number of newly upregulated **FIC1** cells is stored in y_{13} .
- **FIC0** infected cells are upregulated. The infected cells that live in upregulated cubes must have their avidity values and other parameters reset. As we loop through all the **FIC0** cells, if they are in an upregulated cube, we update the avidity value by multiplying the current avidity value by the upregulation factor. If this exceeds the maximum allowable amount, the avidity factor is set to the maximum.
- **FIC1** infected cells are upregulated in the same fashion.
- the amount of collateral damage is then estimated as y_{14} . Recall y_5 is the number of total new infections. The total number of new T cell triggered deaths is the sum $\alpha = \text{new_FIC0_kills} + \text{new_FIC1_kills}$. We can then find the ratio $\alpha = (\text{new_FIC0_kills} + \text{new_FIC1_kills}) / y_5$ for this coarse cube. If this ratio is larger than 1, we reset it to 1. Here, the MHC-I upregulation level stored as $u = p_{18} - 1$. We have computed the $\sum M_i$'s for each coarse cube for different choices of u . For our damage estimate, choose the values $\sum M_i$ corresponding to the value of u we have. In addition, we randomly perturb the $\sum_i M_i^k$ using the parameter p_{15} as follows: $\sum_i M_i^k \rightarrow (1 + \epsilon_i p_{15}) \sum_i M_i^k$ where ϵ_i is a random number in $(-1, 1)$. Finally, the variable, **Update Reg [k]**, stores the number of new IFN- γ mediated upregulations for this time interval for the k^{th} coarse cube. If N_c is the size of a coarse cube (here 19,683), then the ratio $\beta = \text{Update Reg [k]} / N_c$ gives the fraction of cells in each coarse cube which are newly upregulated. For each coarse cube, C_k , we can then calculate $\zeta_k = p_7 \alpha (1 + \epsilon_i p_{15}) (\sum M_i^k) \beta$ where, p_7 is the number of cells we estimate are upregulated per IFN- γ signal (here 1–20). This gives an estimate of the number of cells which are lost to collateral damage. Then $\zeta = \sum_k \zeta_k$ gives the total collateral damage in this time step. For typical simulation values, for $\alpha = 0.1$ (i.e. 10 % of the newly infected cells are killed) and $\beta = 0.001$ (i.e. 0.1 % of the cells in a coarse cube are upregulated, we have approximately $\zeta_k = 0.1 \times 20 \times 420 \times 0.01$ or $\zeta = 8.4$ so that $\zeta \approx 4300$. However, this is a dynamic computation which depends on what is happening currently. The collateral damage at this time step is thus stored in $y_{14} = \zeta$.

In summary, we scale the IFN- γ factor for infected cells in upregulated coarse cubes. We can think of these cube addresses as stored in the variables $U_0(t+1)$ and $U_1(t+1)$. We upregulate by looping over all the addresses in $U_0(t+1)$ and $U_1(t+1)$ as described above. The removal of **FIC0** cells is done in the function $\Omega_0(F_0(t+1))$ which returns the number of ruptured **FIC0** cells in $R_0(t+1)$, the number of T Cell mediated lytic events in $K_0(t+1)$ and upregulation addresses in $U_0(t+1)$. This is written mathematically as

$(R_0(t+1), K_0(t+1), U_0(t+1)) = \Omega_0(F_0(t+1))$ which is Equation (14). Similar calculations are carried out for the **FIC1** case using in the function $\Omega_1(F_1(t+1))$ which returns the number of ruptured **FIC1** cells in $R_1(t+1)$, the number of T cell mediated lytic events in $K_1(t+1)$ and upregulation addresses in $U_1(t+1)$,

Equation (15). The upregulation function $\Gamma_0(U_0(t+1))$ then calculates y_{12} which is written mathematically as $y_{12}(t+1) = \Gamma_0(U_0(t+1))$, (Equation (16)). In a similar fashion, the upregulation function $\Gamma_1(U_1(t+1))$ calculates y_{13} , $y_{13}(t+1) = \Gamma_1(U_1(t+1))$, Equation (17). We also remove cells lysed both by T cells and virus infections alone from our master **FIC0** list, F_0 , to generate a new $F_0(t+1)$.

$$y_{12}(t+1) = \Gamma_0(U_0(t+1)), \text{ New upregulated FIC0 cells} \quad (16)$$

$$y_{13}(t+1) = \Gamma_1(U_1(t+1)), \text{ New upregulated FIC1 cells} \quad (17)$$

Cells that are removed due to virus infection only, simply release virus into the extracellular environment of the host. So for these removed cells, virus released from these cells must be added to the total virus present. This amount of new virus added due to the death of **FIC0** cells is the variable y_8 . We determine how much virus is released at this time step by multiplying by the number of removed cells by a random perturbation of the base amount p_{10} . Where infected target cells are lysed by T cell clones before the virus matures, infectious progeny virus is not released. At the start of this time step, the size of the **FIC0** list is $|F_0|$. So, since y_2 is the total number of **FIC0** cells, we subtract both the number of cells that die naturally due to virus infection and the cells lysed by T cells to get the current value. The equation needed to do this is

$y_2(t+1) = |F_0(t+1)| - (R_0(t+1) + K_0(t+1))$ which is Equation (19). A similar set of computations is used to remove virus and T cell-lysed **FIC1** cells. However, there is a difference here. The virus grows in **FIC1** cells faster by a factor of p_9 . This is done in the y_0 update equation we discussed earlier using the value computed with the y_9 equation from the previous time step expressed mathematically

$y_9(t+1) = (p_{10} + \rho(p_{11}))R_1(t+1)$, which is Equation (20). We then update the **FIC1** population using

$y_3(t+1) = |F_1(t+1)| - (R_1(t+1) + K_1(t+1))$ Equation (21). Then since y_{10} is an estimate of the number of infected cells recognized at this time step, we calculate its updated value using the equation

$y_{10}(t+1) = K_0(t+1) + K_1(t+1)$ as shown in Equation (22).

$$y_8(t+1) = (p_{10} + \rho(p_{11}))R_0(t+1), \text{ New virus FIC0 natural death} \quad (18)$$

$$y_2(t+1) = |F_0(t+1)| - (R_0(t+1) + K_0(t+1)), \text{ Updated FIC0 pop.} \quad (19)$$

$$y_9(t+1) = (p_{10} + \rho(p_{11}))R_1(t+1), \text{ New virus FIC1 natural death} \quad (20)$$

$$y_3(t+1) = |F_1(t+1)| - (R_1(t+1) + K_1(t+1)), \text{ Updated FIC1 pop.} \quad (21)$$

$$y_{10}(t+1) = (K_0(t+1) + K_1(t+1)), \text{ New recognized infected cells} \quad (22)$$

When infected cells are recognized, T cells undergo clonal expansion which improves the T cell killing percentage. Hence, it seems reasonable to allow the parameter p_8 to be reset each time step. To model clonal expansion, we keep track of the fraction of new T cells using the ratio f_t defined by

$$f_t = \frac{\text{number of new T cells}}{\text{current total population of T cells}} = \begin{cases} \frac{y_{10}(t+1)}{y_4(t+1)} & y_{10}(t+1) \geq 1, y_4(t+1) \geq 1 \\ 0.0 & \text{otherwise.} \end{cases} \quad (23)$$

The parameter $p_8(t+1)$ is then reset where we also set up a ceiling threshold which p_8 can not go past, p_8^{\max} . We update with $p_8(t+1) = \min((1+f_t)p_8(t), p_8^{\max})$. If the T Cells which have spread avidity have been cloned, then there is a concomitant B cell clonal expansion which increases the probability that free virus will be neutralized. This probability is stored in the variable p_3 . To implement this idea, we will update p_3 as follows: calculate $g_t = (1+f_t)p_8(t)$ and then do the reset $p_3(t+1) = \min(g_t p_3(t), p_3^{\max})$ where we make sure we do not exceed a specified maximum. Finally, we look at the Uninfected Cell population dynamics:

uninfected cells at next time step = **old uninfected cells** + **uninfected cells homeostasis update** – **new infected cells** – **cells lost due to T cell lysis of uninfected cells, i.e. cells lost due to collateral damage**. We assume the uninfected cells follow a logistics growth model in the absence of infection, $y_1' = p_5 y_1 (p_6 - y_1)$, where the values of the growth factor p_5 and the resource limitation p_6 are chosen appropriately. We thus generate Equation (25) which is our estimate of *healthy* cells in the simulation model which uses our collateral damage estimate from Equation (24).

$$y_{14}(t+1) = \sum_k C_k(t+1) \quad (24)$$

$$y_1(t+1) = y_1(t) + p_5 y_1(t)(p_6 - y_1(t)) - y_5(t) - y_{14}(t+1) \quad (25)$$

6. Damage Due to the Decoy Hypothesis

We are now in a position to explain the collateral damage due to the decoy model. In addition, it is known that there have been puzzling results reported in survival experiments for WNV infections. Recall, the amount of virus involved in an infection is measured in plaque forming units (pfu). This determines the concentration of infectious virus by virtue of the number of areas of cell death in a cell monolayer *in vitro*. Our simulations use a parameter in our simulations for the pfu level and, of course, it is difficult to calibrate *artificial pfu* levels to measured data levels as the *pfu* measure itself is a large scale phenomenon and so it is complicated to model. A typical survival simulation generates the data shown in **Table 5** for an upregulation level of 9. We model host infections in 18 groups using initial viral concentrations ranging from 100 to 1.2e+06 pfu. Note there is

Table 5. Sample simulation data.

Surviving hosts	PFU level	HC ratio	Surviving hosts	PFU level	HC ratio
10	100	0.999990	4	25,000	0.439213
10	250	0.999982	2	50,000	0.358584
10	500	0.808539	2	75,000	0.367035
8	750	0.589315	3	90,000	0.379991
10	1000	0.739119	4	120,000	0.359158
8	2500	0.497502	0	360,000	0.296874
5	5000	0.454120	1	600,000	0.301097
3	7500	0.425226	2	900,000	0.356034
3	10,000	0.370141	0	1,200,000	0.297576

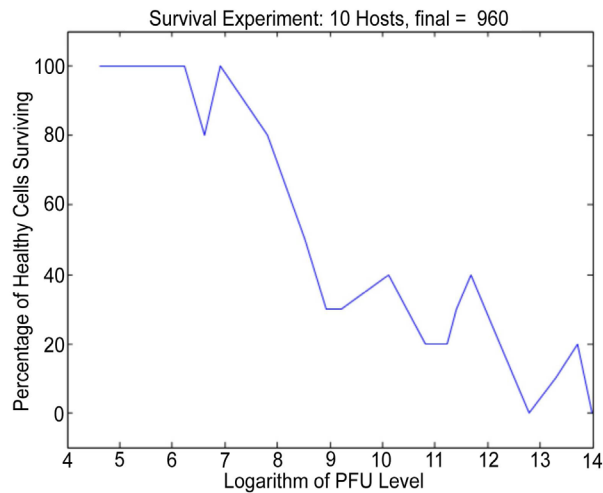


Figure 2. The percentage of uninfected cells vs. viral infection level.

upward movement in the healthy cell ratio even as the pfu level increases which is an indication that the collateral damage model we have discussed here due to the decoy model is a useful explanatory tool. We can also plot the average uninfected cell percentage vs the logarithm of virus concentration (pfu) in **Figure 2**. The spikes in the numbers of surviving hosts with increasing inoculating virus concentration indicate that our simulation validates what we would expect to happen if the decoy model holds true.

7. Conclusion

We show that the decoy hypothesis is a reasonable way to explain collateral damage. Our simulations also suggest that the decoy hypothesis might provide an explanation for survival curve data measured in WNV infections which show an increase in survivability even with increasing viral dose. We will explore that further in other work.

References

- [1] Abbas, A.K., Lichtman, A.H. and Pillai, S. (2010) Cellular and Molecular Immunology. Saunders Elsevier, Philadelphia.
- [2] King, N.J.C. and Kesson, A.M. (1988) Interferon-Independent Increases in Class I Major Histocompatibility Complex Antigen Expression Follow Flavivirus Infection. *Journal of General Virology*, **69**, 2535-2543. <http://dx.doi.org/10.1099/0022-1317-69-10-2535>
- [3] Douglas, D.W., Kesson, A.M. and King, N.J.C. (1994) CTL Recognition of West Nile Virus-Infected Fibroblasts Is Cell Cycle Dependent and Is Associated with Virus-Induced Increases in Class I MHC Antigen Expression. *Immunology*, **82**, 561-570.
- [4] Kesson, A.M., Cheng, Y. and King, N.J.C. (2002) Regulation of Immune Recognition Molecules by Flavivirus, West Nile. *Viral Immunology*, **15**, 273-283. <http://dx.doi.org/10.1089/08828240260066224>
- [5] Müllbacher, A., Hill, A.B., Blanden, R.V., Cowden, W.B., King, N.J.C. and Hla, R.T. (1991) Alloreactive Cytotoxic T Cells Recognize MHC Class I Antigen without Peptide Specificity. *Journal of Immunology*, **147**, 1765-1772.
- [6] King, N.J.C., Mullbacher, A., Tian, L., Rodger, J.C., Lidbury, B. and Hla, R.T. (1993) West Nile Virus Infection Induces Susceptibility of *in Vitro* Outgrown Murine Blastocysts to Specific Lysis by Paternally Directed Allo-Immune and Virus-Immune Cytotoxic T Cells. *Journal of Reproductive Immunology*, **23**, 131-144. [http://dx.doi.org/10.1016/0165-0378\(93\)90003-Z](http://dx.doi.org/10.1016/0165-0378(93)90003-Z)
- [7] King, N.J.K., Davison, A., Getts, D., Ping Lu, D., Teague Getts, M., Yeung, A., Peterson, J. and Kesson, A.M. (2008) Enhanced Antigen Processing or Immune Evasion? West Nile Virus and the Induction of Immune Recognition Molecules. In: Diamond, M.S., Ed., *West Nile Encephalitis Virus Infection: Viral Pathogenesis and the Host Immune Response, Emerging Infections in the 21st Century*, Springer-Verlag, Heidelberg, 309-339.
- [8] Kuhlman, P., Moy, V.T., Lollo, B.A. and Brian, A.A. (1991) The Accessory Function of Murine Intercellular Adhesion Molecule-1 in T Lymphocyte Activation. Contributions of Adhesion and Co-Activation. *Journal of Immunology*, **146**, 1773-1782.
- [9] Shen, J., T-To, S.S., Schrieber, L. and King, N.J.C. (1997) Early E-Selectin and VCAM-1 and ICAM-1 and Late Major Histocompatibility Complex Antigen Induction on Human Endothelial Cells by Flavivirus and Comodulation of Adhesion Molecule Expression by Immune Cytokines. *Journal of Virology*, **71**, 9323-9332.
- [10] Getts, D.R., Matsumoto, I., Muller, M., Getts, M.T., Radford, J., Shrestha, B., Campbell, I.L. and King, N.J.C. (2007) Role of IFN- γ in an Experimental Murine Model of West Nile Virus-Induced Seizures. *Journal of Neurochemistry*, **103**, 1019-1030. <http://dx.doi.org/10.1111/j.1471-4159.2007.04798.x>
- [11] Peterson, J.K., King, N.J.C. and Kesson, A.M. (2014) Modeling West Nile Virus One Host Infections. Technical Report, Clemson University, Clemson. <http://www.ces.clemson.edu/%7Epetersj/CurrentPapers/WNVSimulationCode-04082014.pdf>
- [12] Alon, U. (2006) An Introduction to Systems Biology: Design Principles of Biological Circuits. CRC Mathematical and Computational Biology, Chapman Hill.
- [13] King, N.J.C. and Kesson, A.M. (2003) Interaction of Flaviviruses with Cells of the Vertebrate Host and Decoy of the Immune Response. *Immunology and Cell Biology*, **81**, 207-216. <http://dx.doi.org/10.1046/j.1440-1711.2003.01167.x>
- [14] Lobigs, M., Mullbacher, A. and Regner, M. (2003) MHC Class I Up-Regulation by Flaviviruses: Immune Interaction with Unknown Advantage to Host or Pathogen. *Immunology and Cell Biology*, **81**, 217-223. <http://dx.doi.org/10.1046/j.1440-1711.2003.01161.x>

Scientific Research Publishing (SCIRP) is one of the largest Open Access journal publishers. It is currently publishing more than 200 open access, online, peer-reviewed journals covering a wide range of academic disciplines. SCIRP serves the worldwide academic communities and contributes to the progress and application of science with its publication.

Other selected journals from SCIRP are listed as below. Submit your manuscript to us via either submit@scirp.org or **Online Submission Portal**.

