

# C-ImmSim: a model of immunological models

Filippo Castiglione  
Institute for Computing Applications “M.Picone”  
National Research Council (CNR) of Italy  
[f.castiglione@iac.cnr.it](mailto:f.castiglione@iac.cnr.it)

## Abstract

This document provides a short introduction to the immunology included in the C-ImmSim model plus few examples showing the behavior of C-ImmSim with respect to differently modeled pathogens.

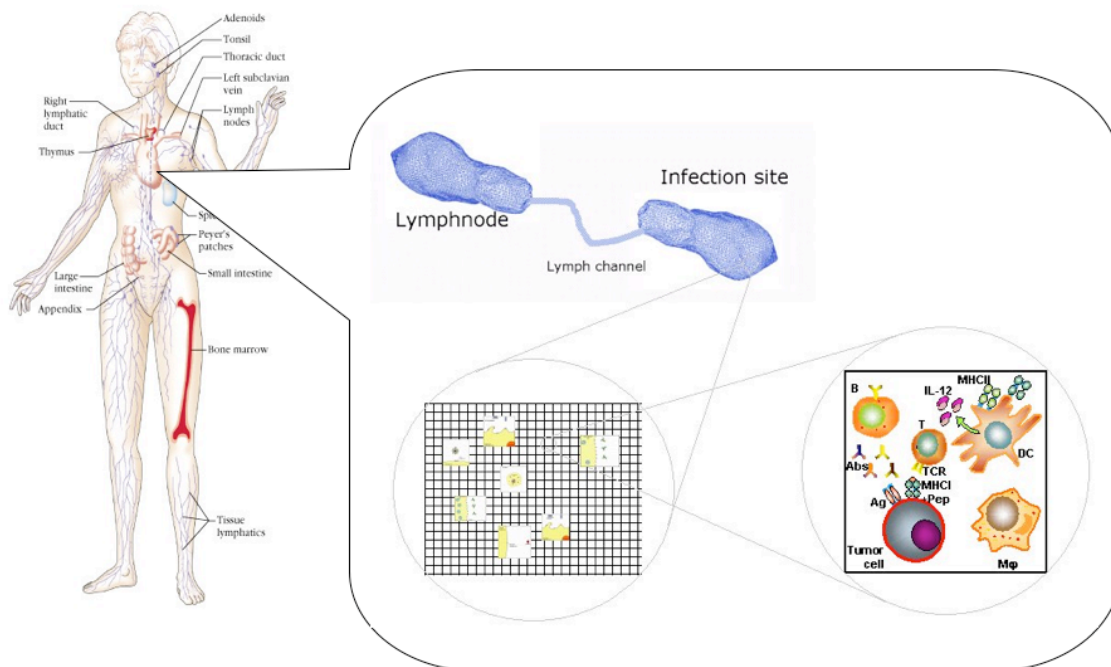
## Index

C-ImmSim a model of immunological models .....	1
Immune description (C-ImmSim) .....	2
Affinity .....	3
Homeostasis .....	3
Peptide digestion and presentation .....	3
Immune memory .....	4
Clonal selection theory .....	4
Thymus education of T lymphocytes (clonal deletion theory) .....	5
Hypermutation of antibodies .....	5
Hayflick limit (T cells replicative senescence) .....	5
Anergy .....	5
Matzinger’s danger signals .....	6
Idiotypic Network theory .....	6
References .....	7
References on IMM SIM and C-ImmSim .....	8
Example of how C-ImmSim adheres to major immune functions .....	9
Primary and secondary responses .....	10
Immune Memory .....	12
Affinity maturation .....	13
Self/non-self discrimination .....	15
Bacterial growth .....	18
Viral infection .....	20
HIV infection .....	22

## Immune description (C-ImmSim)

The C-ImmSim model describes both the *humoral* and the *cellular* response of a mammalian immune system to the presence of antigens (virus, bacteria, etc.), at the *cellular level* (mesoscopic scale)

One or more cubic millimeters of a secondary lymphoid organ (or, at convenience, micro litres of peripheral blood) of a vertebrate animal is mapped onto a three-dimensional lattice with periodic boundary conditions.



**Figure 1 C-ImmSim simulates a portion of a lymph node. A more recent version of the code allows defining arbitrary meshes to represent more than one organ.**

A time step of the simulation corresponds to few hours of real life. The user can choose to have 8, 4, 2 or 1 hour as a definition of a simulated time step.

C-ImmSim is an Agent-Based Model (ABM, [1]) hence cells are represented as agents, i.e., they keep their individual experience throughout their simulated life span. Cells interact locally (i.e., inside a lattice site).

Major classes of cells of the lymphoid lineage (lymphocytes T helper, TH, and cytotoxic, CTL, lymphocytes B and deriving antibody-producing plasma cells, PLB) and some of the myeloid lineage (macrophages, MA, and dendritic cells, DC) are considered.

C-ImmSim is a polyclonal model since an arbitrary repertoire size of lymphocytes can be represented (as opposed to the monoclonal models where only a single

population of genetically identical lymphocytes is represented). To be more precise, by defining NBITS the length of the binary strings representing the cell receptors or antibodies, the potential repertoire size is  $2^{\text{NBITS}}$ .

Lymphocytes are created in the bone marrow compartment. Their receptor is randomly chosen in the bit string space  $\{0,1\}^{\text{NBITS}}$ . While B lymphocytes goes into circulation directly, the T lymphocytes pass through the Thymus where they undergo selection for auto reactive cells. This step requires the definition of what the “self” is (explained below).

The mobility of cells is modelled by taking into account realistic diffusion coefficients observed in vivo. T cells have faster diffusion constant than B cells [2][3]. Moreover, in absence of chemotaxis, macrophage and dendritic cells have diffusivity similar to B cells [4]. Molecules, instead, (like IL2, D signal, etc.) diffuse much faster than both T and B cells but their effective displacement and interaction capability is limited by their half-life and their concentration.

Chemotaxis is modeled in a quite simple way, that is, by having the cells to move (on average) in the direction of the higher gradient of the chemotactic agent  $c(x,t)$  at position  $x$  at time  $t$ . Chemokines are represented as continuous fields and therefore

$$\frac{\partial c(x,t)}{\partial t} = D\nabla^2 c(x,t)$$

their diffusion is governed by a simple heat equation with a diffusion coefficient  $D$

### **Affinity**

Two bit-strings complement each other (or are a perfect match) if every 0 in one corresponds to a 1 in the other and conversely. More generally, an  $m$ -bit match is defined as a pair where exactly  $m$  bits complement each other. The affinity is then defined as a monotonic function of  $m$ ; higher  $m$  the more likely for the two strings to bind. The affinity also includes a cut-off; for matching smaller than this threshold value, no binding occurs.

### **Homeostasis**

Homeostasis is implemented by a means of a mean reverting process. This is used to calculate how many cells, for each cell type, are to be created at each time step, or, how many cells have to be eliminated. The actual deletion of the cells from the simulation is made stochastically by taking into account their individual half-life.

### **Peptide digestion and presentation**

C-ImmSim implements peptide digestion and presentation of peptides on both MHC class I and class II molecules (exogenous and endogenous pathways). Since MHC molecules are also represented as binary strings, the MHC-peptide binding is also modeled through a function of the matching bits in the bit string comparison.

## **Immune memory**

While researchers completely agree on the importance of some invoked mechanisms (increased number of specific B and T cells available, persistence of activated cells), on others, especially those invoked to justify long-lasting memory (persistence of antigen, continuous stimulation, idiotypic network, long-living memory cells), the consensus is far from complete.

In C-ImmSim we consider the memory as a cell's state acquired during active participation to successive (and successful) immune responses. We model memory of lymphocytes by a mechanism that increases the half-life of a cell by a certain amount every time that cells participate in a successful interaction. The rationale behind this modeling choice is that useful cells survive longer than useless simply because they get a whole lot of stimulations during the immune reactions.

The overall result of this process is that few cells increase their half-life considerably and live longer than other cells. Moreover, in this way we obtain an expansion of the memory compartment that is somehow proportional (in a non linear way, to be correct) to the magnitude of the infection and consequent duration of the immune response.

On top of the “basic” mechanisms of the cellular dynamics described so far, C-ImmSim incorporates quite a few working assumptions/theories (some of them can be easily toggled on/off, others require a complete rethinking of the model). These are:

- Clonal selection theory
- Thymus education of T lymphocytes (clonal deletion theory)
- Hypermutation of antibodies
- Hayflick limit (T cells replicative senescence)
- Anergy
- T cell anergy
- Ag-dose induced tolerance (anergy) in B cell
- Matzinger's danger signals
- Idiotypic Network theory

### **Clonal selection theory**

In 1954, immunologist Niels Jerne put forward a theory, which stated that there is already a vast array of lymphocytes in the body prior to any infection. The entrance of an antigen into the body results in only one type of lymphocyte to match it and produce a corresponding antibody to destroy it. This selection of only one type of lymphocyte results in it being cloned or reproduced by the body extensively to ensure there are enough antibodies produced to inhibit and prevent infection. By representing a vast repertoire, C-ImmSim obeys the clonal selection theory.

## **Thymus education of T lymphocytes (clonal deletion theory)**

The clonal Deletion theory, proposed by Burnet, according to which self-reactive lymphoid cells are destroyed during the development of the immune system in an individual. For their work Frank M. Burnet and Peter B. Medawar were awarded the 1960 Nobel Prize in Physiology or Medicine “for discovery of acquired immunological tolerance”.

(*Positive selection*) Cells that do not express a functional TCR (T cell receptor), or a TCR without affinity for self-MHC molecules, will undergo apoptosis (death by neglect), whereas cells with TCRs of high affinity towards self-peptide / self-MHC molecule complexes are eliminated by acute apoptosis (*Negative selection*).

Only the appropriate (intermediate) level of TCR signaling initiates positive selection and maturation, either into CD4+CD8- or CD4-CD8+ cells when selected on class II or class I MHC molecules, respectively. Mature T cells leave the thymus to populate the peripheral lymphoid organs.

## **Hypermutation of antibodies**

Duplicating B cells have a probability to mutate their receptor. The probability to flip a bit has been calculated in [5]

## **Hayflick limit (T cells replicative senescence)**

The Hayflick Limit Theory of Aging suggests that the human cell is limited in the number of times it can divide. Working with Dr. Moorehead in 1961, Dr. Hayflick theorized that the human cells ability to divide is limited to approximately 50-times, after which they simply stop dividing (and hence die) [6][7][8]

Studies have firmly established that T cells are similar to other normal human somatic cells in that they invariably undergo replicative senescence in cell culture [9]. In summary, in C-ImmSim, the number a T helper replicates is limited to 50.

## **Energy**

Energy is a theory that describes a lack of reaction by the body's defense mechanisms to foreign substances, and consists of a direct induction of peripheral lymphocyte tolerance. It is a state of immune unresponsiveness.

## **T cell anergy**

T cell anergy is induced when the T cell's antigen receptor is stimulated, effectively freezing T cell responses pending a “second signal” from the antigen-presenting cell. The delivery of the second signal by the antigen-presenting cell rescues the activated T cell from anergy, allowing it to produce the lymphokines necessary for the growth of additional T cells [10].

In practice in C-ImmSim we implement the following two rules:

- a. TH need a first activation by APCs (MA+DC) otherwise they go anergic upon contact with B cells
- b. TC need a first activation by infected cells (MA+EP) otherwise they go anergic upon contact any other infected cell.

### **Ag-dose induced tolerance (anergy) in B cell**

Tolerance is an active immunologic process that is mediated by more than one mechanism and is dose-dependent. Low doses of antigen administration favor the induction of active cellular regulation, whereas higher doses favor the induction of clonal anergy [11][12]. In practice, in C-ImmSim, the rule is: High doses of antigen induce B cell anergy

### **Matzinger's danger signals**

In a 1994 article entitled "Tolerance, Danger and the Extended Family", Matzinger lay out the idea that antigen-presenting cells respond to "danger signals" - most notably from cells undergoing injury, or stress or "bad cell death" (as opposed to apoptosis, controlled cell death). The alarm signals released by these cells let the immune system know that there is a problem requiring an immune response. She argued that T-cells and the immune response they orchestrate do occur on a dynamic and constantly updated response to danger as defined by cellular damage [13].

In practice in C-ImmSim, this is translated in the following two rules:

- a. Cells undergoing "bad cell death" (because of virus infection or lysis by cytotoxic cells) release danger signal molecules;
- b. Macrophages are activated by danger signaling molecules.

### **Idiotypic Network theory**

C-ImmSim allows the user to define the antibody's paratope (Fc) as a binary string. Therefore one can simulate the Idiotypic Network theory of Jerne [14]. Fc represents the constant region of the immunoglobulin molecule that can be seen by the immune system as an antigen on its own. Since the antibodies peptides can be phagocytosed as any other molecule by the APCs, hence digested and presented to the THs, they can elicit an immune response on their own. As a consequence of this fact, a chain of recognition can take place that keeps the immune system stimulated all the time. According to Jerne's theory, immunization with an antigen may lead to the generation not only of antigen-specific antibodies (Ab1), but also to antibodies that recognize Ab1. This occurs because the unique structure (idiotype) of the antigen-binding site of Ab1 stimulates the immune system to generate Ab2, which mimics the structure of the antigen. Eventually, a similar mechanism generates Ab3. Ab1 and Ab3 have similar binding capacities and, in many cases, identical amino acid sequences at their antigen-binding sites. As also proposed by Jerne, a unique

structure (idiotype) in the framework of Ab1 can trigger the idiotypic network (from [15]).

## References

- [1] Castiglione F. (2006) Agent Based Modeling. Scholarpedia, p.3735, [http://www.scholarpedia.org/article/Agent\\_Based\\_Modeling](http://www.scholarpedia.org/article/Agent_Based_Modeling).
- [2] Mark J. Miller, Sindy H. Wei, Michael D. Cahalan, and Ian Parker. Autonomous t cell trafficking examines in vivo with intravital two-photon microscopy. PNAS, 100:2604–2609, 2003.
- [3] Mark J. Miller, Sindy H. Wei, Ian Parker, and Michael D. Cahalan. Two-photon imaging of lymphocyte motility and antigen response in intact lymph node. Science, 296:1869–1873, 2002.
- [4] Philippe Bousso and Ellen Robey. Dynamics of cd8+ t cell priming by dendritic cells in intact lymph nodes. Nature Immunology, 4:579–585, 2003.
- [5] F. Celada and P.E. Seiden. Affinity maturation and hypermutation in a simulation of the humoral immune response. Eur. J. Immunol. 1996; **26**: 1350-.
- [6] Jerry W. Shay and Woodring E. Wright, Hayflick, his limit, and cellular ageing. Nature Review, Molecular and Cell Biology. 2000;1:72-76.
- [7] Hayflick L. The limited in vitro lifetime of human diploid cell strains. Exp Cell Res. 1965;37:614–636.
- [8] Hayflick L, Moorhead PS. The serial cultivation of human diploid cell strains. Exp Cell Res. 1961;25:585–621.
- [9] Effros RB, Pawelec G. Replicative senescence of T lymphocyte: Does the Hayflick limit lead to immune exhaustion? Immunology Today. 1997;18:450–454.
- [10] Ronald H. Schwartz. T cell anergy. Annual Review of Immunology, 2003;21: 305-334.
- [11] Nossal GJV, Pike Beverley L. Clonal anergy: Persistence in tolerant mice of antigen-binding B lymphocytes incapable of responding to antigen or mitogen. Proc Natl Acad Sci U S A. 1980 Mar;77(3):1602–1606.
- [12] Whitacre CC, Gienapp IE, Orosz CG, Bitar D. Oral tolerance in experimental autoimmune encephalomyelitis. III. Evidence for clonal anergy. J Immunol. 1991;147:2155–2163.
- [13] Matzinger, P. (1994). Tolerance, Danger, and the Extended Family. Ann. Reviews of Immunology, 12, 991-1045.
- [14] Jerne, N. K., 1974, Towards a network theory of the immune system, Ann. Immunol. 125:373–389.
- [15] Yehuda Shoenfeld Y. The idiotypic network in autoimmunity: antibodies that bind antibodies that bind antibodies. Nature Medicine 10, 17–18 (2004).

## References on IMMSIM and C-ImmSim

- Celada F, Seiden P. *Immunology Today* 1992; **13**:56.
- Seiden P, Celada F. *J. Theor. Biol.* 1992; **158**:329.
- Morpurgo D, Serenthà R, Seiden PE, Celada F. *International Immunology* 1995; **7**:505.
- Celada F, Seiden P. *Eur. J. Immunol.* 1996; **26**:1350.
- Bezzi M, Celada F, Ruffo S, Seiden PE. *Physica A*, 1997; **245**:145.
- Kohler B, Puzone R, Seiden P, Celada F. *Vaccine* 1999; **19**:862.
- Succi S, Castiglione F, Bernaschi M. *Phys Rev Lett*, 1997; **79**:22 4493
- Bernaschi M, Castiglione F, Succi S. *Phys. Rev. E* 2000; **61**:1851.
- Bernaschi M, Castiglione F. *Computers in Biology and Medicine* 2001; **31**:303.
- Castiglione F, Sleitser V. Agur Z. in *Cancer Modeling and Simulation*, Preziosi L. (ed.), Chapman & Hall/CRC Press (UK), London, June 2, 2003;
- Bernaschi M, Castiglione F. *Immunology and Cell Biology* 2002; **80**, 307.
- F. Castiglione, F. Poccia, G. D'Offizi and M. Bernaschi. *AIDS Res Hum Retrovirus* 2004; **20(12)**, 1316-1325.
- F. Castiglione and Bernaschi M. *Int J Mod Phys C* 2006; **16(12)**, 1869-1879.
- V. Baldazzi, F. Castiglione and M. Bernaschi. *Cellular Immunology* 2006; **244**, 77-79.
- F. Castiglione, K.A. Duca, A. Jarrar, R. Laubenbacher, K. Luzuriaga, D. Hochberg and D.A. Thorley-Lawson. *Bioinformatics* 2007; doi: 10.1093/bioinformatics/btm044
- F. Castiglione, F. Pappalardo, M. Bernaschi, S. Motta. *Bioinformatics* 2007, to appear.
- F. Pappalardo, P.-L. Lollini, F. Castiglione, S. Motta. *Bioinformatics* 2005; **21(12)**:2891-7.
- F. Castiglione, F. Toschi, M. Bernaschi, S. Succi, R. Benedetti, B. Falini and A. Liso. *J Theo Biol* 2005; **237/4**: 390-400.



## **Example of how C-ImmSim adheres to major immune functions**

Here below we present examples of simulations that show specific characteristics of the model. For each simulation experiment the user has to modify the “standard” input parameter set called `datPlain_12`. In the following, he can find the specific instructions on how to accomplish this task.

DRAFT

## Primary and secondary responses

The first goal of any model of the immune system is to reproduce, at least qualitatively, a primary and a secondary immune response to the same antigen injected twice.

While primary immunization (primary response) is relatively slow (4-7days), small (low concentration of antibody) and consisting of low affinity antibody (IgM first, IgG second), the secondary immunization (secondary response or booster immunization) is relatively fast (2-4 day), with large amounts of high affinity antibody, mostly IgG.

Before starting the experiment we give a look at the input parameter file `datPlain_12`.

- Note that the epitope in the `datPlain12_exp1` is 0 that is the binary string 0000 0000 0000.
- Therefore the perfect match antibody is the string 1111 1111 1111.
- Also note that the `minmatch = 9` that means that a mismatch = 3 is the minimum allowed to start an immune response.

Therefore we have

- 12 one bit mismatch (e.g., 011111111111, or 101111111111),
- 66 two bits mismatch (e.g., 111100111111 or 011111111011),
- and 220 three bits mismatch (e.g., 00110111111, 010111110111, 11011110011).

For this experiment, no modification to the `datPlain_12` file is necessary.

By running the simulation we obtain the plots shown in figure 2. To plot clone-specific information, execute “extract” to uncompress the binary file `_?`, `_??`, and `_???` (follow the instructions given in the documentation). This utility creates a set of files of the type `_.dat`, and `_.plt`; the first is the data, the second is a gnuplot script to plot the data. In addition to that, the simulation produces files, named `_*-details.out`, containing cumulative info regarding cell-state info, one file for each cell type. These can be visualized/plotted by using the gnuplot scripts (found in the `../plt/Plain/` directory) provided with the package.

Those plot show that the secondary immune response is faster and stronger than the first (left panel). The same figure (right panel) shows the ten best responder antibodies by clonotypes represented as binary string. It is worth to note that those string have a match with the antigen epitope that is above the threshold defined in the input parameter file.

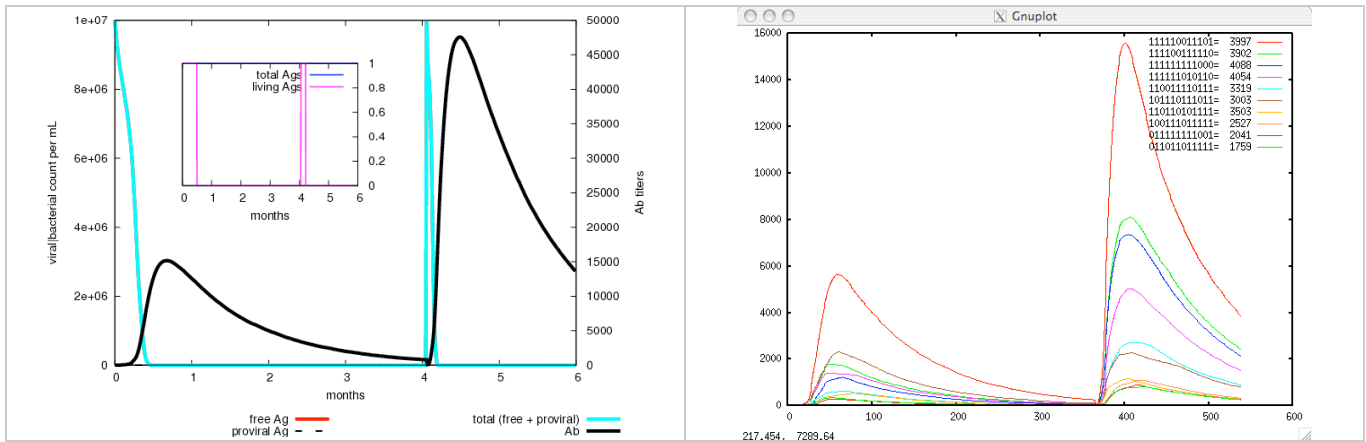


Figure 2 Panel on the left shows the antigen and antibody concentration. That on the right shows the antibodies for clonotypes (only the first ten best responders).

## Immune Memory

The immune memory is achieved through a mechanism of “sustain” that increases the life expectancy of cell that actively cooperate to the immune response (this is explained in a previous paragraph).

In order to show that it works in practice we induce a primary and secondary immune response to one antigen that we inject twice but it mounts a primary response to another different antigen that we inject just once.

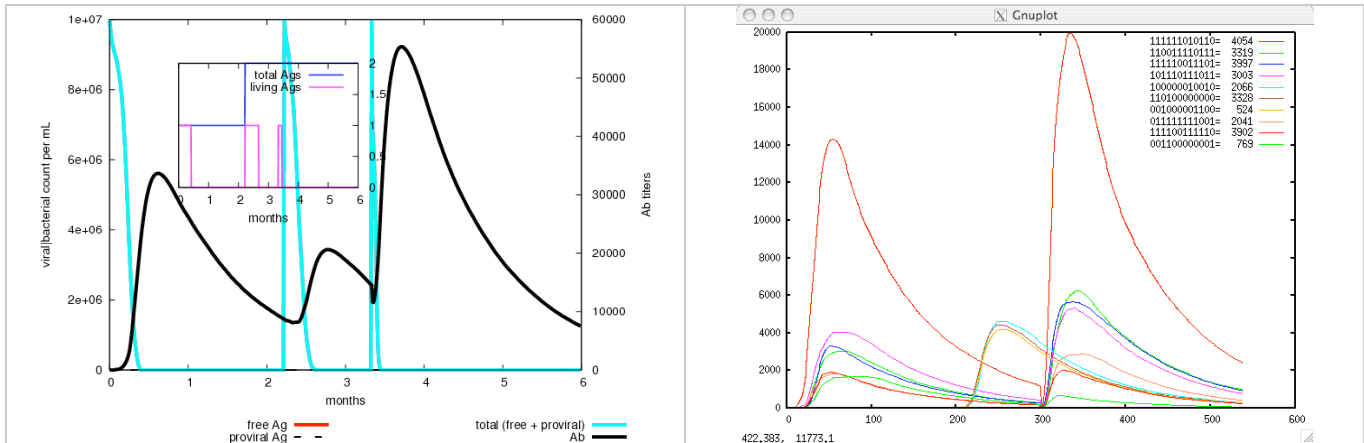
Modify the datPlain\_12 in the following way (parts in red) so to have two different antigens

1. The first injected twice
2. The second injected just once. In other words the sequence of injections is Ag1, Ag2, Ag1.
3. To simplify we use the same peptide (i.e., same affinity to MHC)
4. but different epitope (want to see different humoral response)

```
...
----- Antigen related -----
<3> #inj Ag
<1> Inside (peptide)
<1> Outside (epitope)
----- Antigen injections -----
<0> RandomCarrier
<0> Time to inj ag
<0>  unharful bacteria
<0.0>
<10000> Num Ag to inj ag
<1234> peptide
<0> epitope
<200> Time to inj ag
<0>  unharful bacteria
<0.0>
<10000> Num Ag to inj ag
<1234> peptide
<4095> epitope
<300> Time to inj ag
<0>  unharful bacteria
<0.0>
<10000> Num Ag to inj ag
<1234> peptide
<0> epitope
----- Cell Duplication -----
...
```

Now run the simulation to obtain the plots in figure 3.

The panel on the left shows that in correspondence to the first injection of the first antigen and to the first injection of the second antigen the immune response is a primary one while the antibody peak in correspondence to the second injection of the first antigen is really a secondary immune response due to the existing immunological memory.



**Figure 3 Three injections; two of the same antigen and one of a different antigen. The different level of response shows that the immune system has mounted an immune memory to the antigen injected twice (compare the binary strings for matching with the injected antigen peptides).**

### Affinity maturation

Since C-ImmSim is a polyclonal model that reproduces the immunological memory, it is equipped with the hypermutation and obeys the laws of the theory of clonal selection. Successive expositions to the same antigen will induce the maturation of better affinity clones.

In order to better evidence the maturation of higher affinity we use a parameter, called “hole”, that creates an artificial hole in the repertoire deleting high matching clones directly at creation time. Moreover, to amplify the effects we simulate a chronic infection with a continuous antigenic stimulus.

### Copy the datPlain12 in datPlain12\_exp3 and

1. modify the number of time-steps of the simulation,
2. modify the number of injected antigens to a smaller dose,
3. modify it so to have one injection at each time step.

```

...
----- Number of time steps-----
<1000> num_step /* a year is 1095 */
...
----- Antigen related -----
<1> #inj Ag

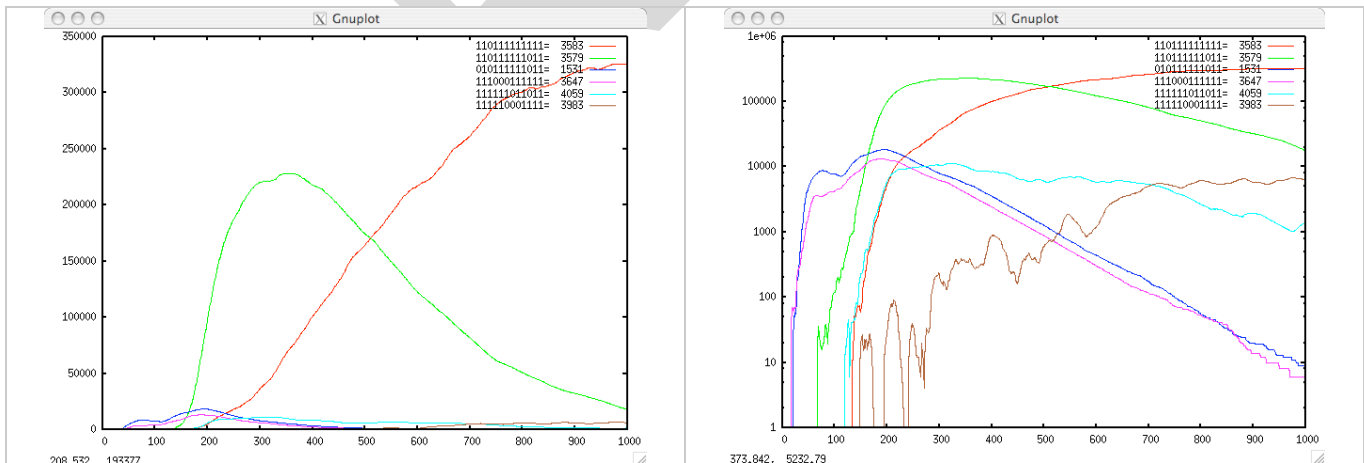
```

```

<1> Inside (peptide)
<1> Outside (epitope)
----- Antigen injections -----
<0> RandomCarrier
<0> Time to inj ag
<0>          unharmed bacteria
<0.0>
<500> Num Ag to inj ag
<2415> peptide
<0> epitope
----- Cell Duplication -----
<4> DupStep_B
<4> DupStep_TH
<4> DupStep_TC
----- Various Flags -----
<1> Cont_Inj_Flag {0, 1}
----- HIV related -----
...

```

Run the simulation and look at the plots in figure 4. The panel on the right is the same as the left one but in log scale. The figure shows the appearance of best matching clones while the time passes. Those supersede the low matching ones in the long run thanks to the continuous availability of antigens (remember it is a chronic infection) that reduces the effect of competition among clones (otherwise we would not have been able to see the maturation due to the competition among clones to capture the antigen).



**Figure 4 Panel on the left (right panel is the same but in log scale) shows the appearance of better matching clones. Compare the binary strings in the key legend with the antigen epitope in the datPlain\_12 file.**

## Self/non-self discrimination

C-ImmSim is equipped with the Thymus compartment that selects out self-reactive T lymphocytes. In order to show how this works we define a set of strings as to be the “self” molecules of the simulated individual and then injecting two antigens; one that is close to self and another that is “far” from self.

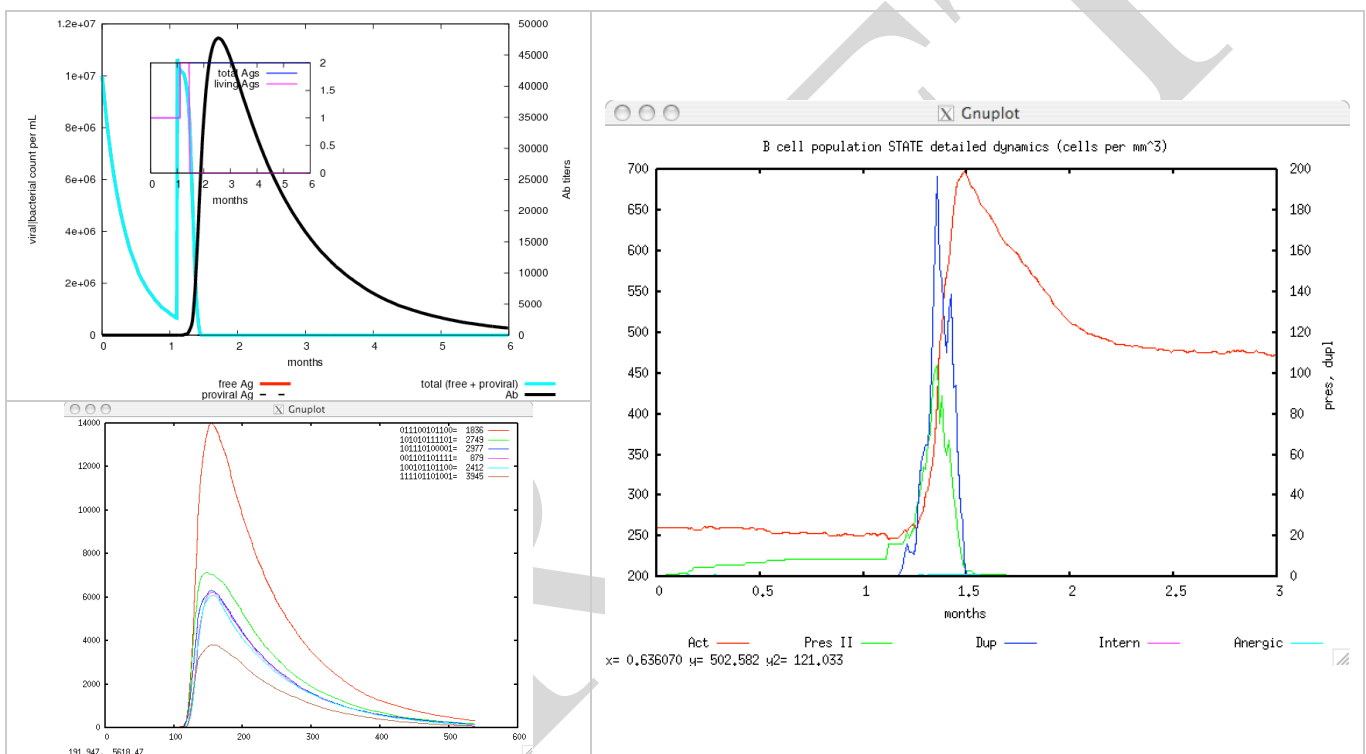
Make a copy of datPlain12 in datPlain\_exp4. Then modify it as follows:

1. change MHC strings;
2. modify the number of self-peptides;
3. specify 12 self peptide strings. For simplicity we define them “close” to the string 000000000000;
4. modify the antigens to inject, one is self, the other one is non-self (the epitope binary string is irrelevant)

```
...
----- MHC molecules -----
<1> NumMHC1      [1, MAX_NUM_MHC1]
<0> MHC1I       [0, 2^NBITSTR-1], -1=random
<1> NumMHCII     [1, MAX_NUM_MHCII]
<0> MHCII1      [0, 2^NBITSTR-1], -1=random
...
----- SelfPeptides & AbPeptides -----
<12> NumSelfPep  [0, NUMSELFPEP]
<0> SelfPep     [0, 2^NBITSTR-1]
<2> SelfPep     [0, 2^NBITSTR-1]
<4> SelfPep     [0, 2^NBITSTR-1]
<8> SelfPep     [0, 2^NBITSTR-1]
<16> SelfPep    [0, 2^NBITSTR-1]
<32> SelfPep    [0, 2^NBITSTR-1]
<64> SelfPep    [0, 2^NBITSTR-1]
<128> SelfPep   [0, 2^NBITSTR-1]
<256> SelfPep   [0, NUMABPEP]
<512> SelfPep   [0, 2^NBITSTR-1]
<1024> SelfPep  [0, 2^NBITSTR-1]
<2048> SelfPep  [0, 2^NBITSTR-1]
<1> NumAbPep    [0, NUMABPEP]
...
<0> Time to inj ag [0, num_step]
<0> unharful bacteria
<0.0>
<10000> Num Ag to inj ag [0, MAX_NUM_TYPE_INIT[_Ag]]
<0> peptide [2600]
<1234> epitope [4095]
<100> Time to inj ag [0, num_step]
<0> unharful bacteria
<0.0>
<10000> Num Ag to inj ag [0, MAX_NUM_TYPE_INIT[_Ag]]
<4095> peptide [2600]
<1234> epitope [4095]
```

...

Execute the code. Since the thymus filters out self-reactive lymphocytes we expect the system not to mount an immune response against the self. This is indeed what happens as shown in figure 5. The panel on the top-left shows a primary response only to the secondly injected antigen that we purposefully chosen among the non-self. In contrast, the first injection does not really induce an immune response (as you can see from the absence of antibodies in panel bottom-left and absence of a clone expansion of B cells in the right panel). The antigen concentration simply decreases because of the slow action of the natural (innate) immune action of antigen processing cells (macrophages and dendritic cells).



**Figure 5** Top-left panel shows a primary response only to the secondly injected antigen that we purposefully chosen among the non-self. In contrast, the first injection does not really induce an immune response: absence of antibodies in panel bottom-left and absence of a clone expansion of B cells in the right panel.



DRAFT

## Bacterial growth

A generic bacterial infection can be easily simulated with C-ImmSim by allowing the antigen to replicate. While specific bacteria may require specific mechanism, the bacterium we simulate is the simplest one and obeys to just one rule: it grows.

The user can define the replication rate in the input parameter file. It is important to observe that the immune system is able to clear the bacterial infection only if this replication rate is not too high. In order to show this fact we perform two (or more) simulations that differ in the replication rate of the injected bacterium.

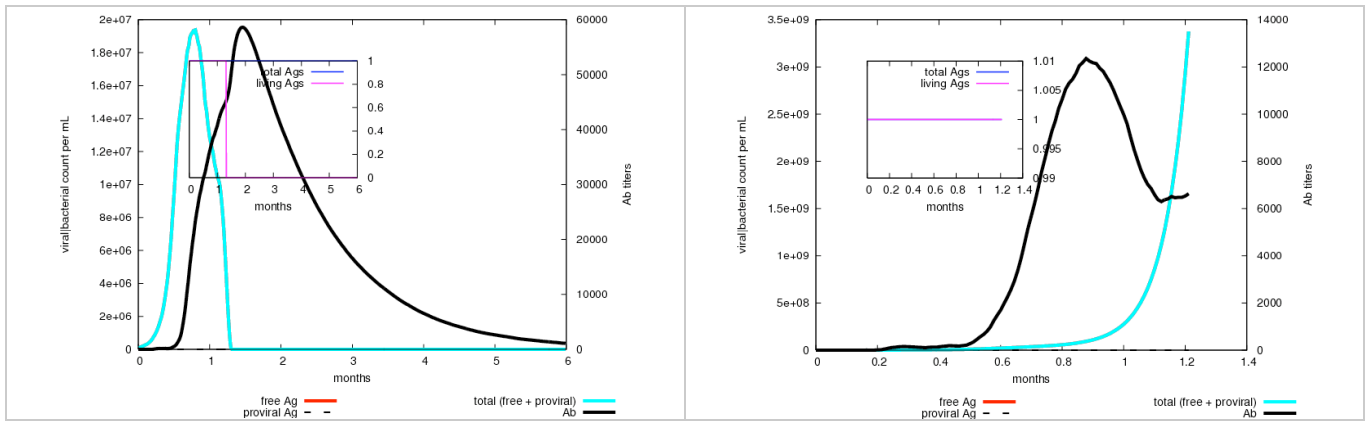
We find that for replication rates below a critical value  $r_c$  the immune system is able to clear the infection, while for replication rates  $> r_c$  the bacterium grows unstopped. We have found the critical value  $0.15445 < r_c < 0.15450$ .

The user is invited to change the `datPlain_12` and to play with the replication rate to verify the threshold value  $r_c$ :

1. Eliminate the second injection;
2. The field at the row just before the "Num Ag to inj ag" (that lacks a comment) is to specify the replication rate "r" of the antigen. If  $r > 0$ , then it is a replicating bacterium.

```
...
----- Antigen related -----
<1> #inj Ag
<1> Inside (peptide)
...
----- Antigen injections -----
<0> RandomCarrier
<0> Time to inj ag
<0> harmful bacteria
<0.15445> repl rate
<100> Num Ag to inj ag
<2415> peptide
<0> epitope
...
```

Figure 6 shows the outcomes of a simulation with  $r < r_c$  (left panel) and  $r > r_c$  (right panel). In the first case the humoral immune response is able to clear the bacteria while in the second case the bacterium replicates too fast for the immune response and cannot be cleared.



**Figure 6** The outcomes of a simulation with the replication rate  $r < r_c$  (left panel) and another simulation with  $r > r_c$  (right panel).

DRAFT

## Viral infection

A viral infection can be easily simulated in C-ImmSim. In this case, since the virus is generic, we use epithelial cells as virus target cells. A viral infection elicits both humoral and cytotoxic immune response.

The user can modify the datPlain\_12 input parameters file to specify both replication rate and the threshold determining the blowing of an infected cell when the viral content reaches it.

1. Eliminate the second injection;
2. change the probability to infect the epithelials;
3. change the number of injections;
4. change the field "what" from 0 (=bacterium) to 1 (=virus);
5. change the replication rate  $r > 0$ ;
6. change the number of viral particles to inject;
7. change the epitope of the virus;
8. change the number of viral particles at "blow time"

...  
<0.1> p\_EP\_Virus

...  
<1> #inj Ag

...  
<1> unharmed bacteria

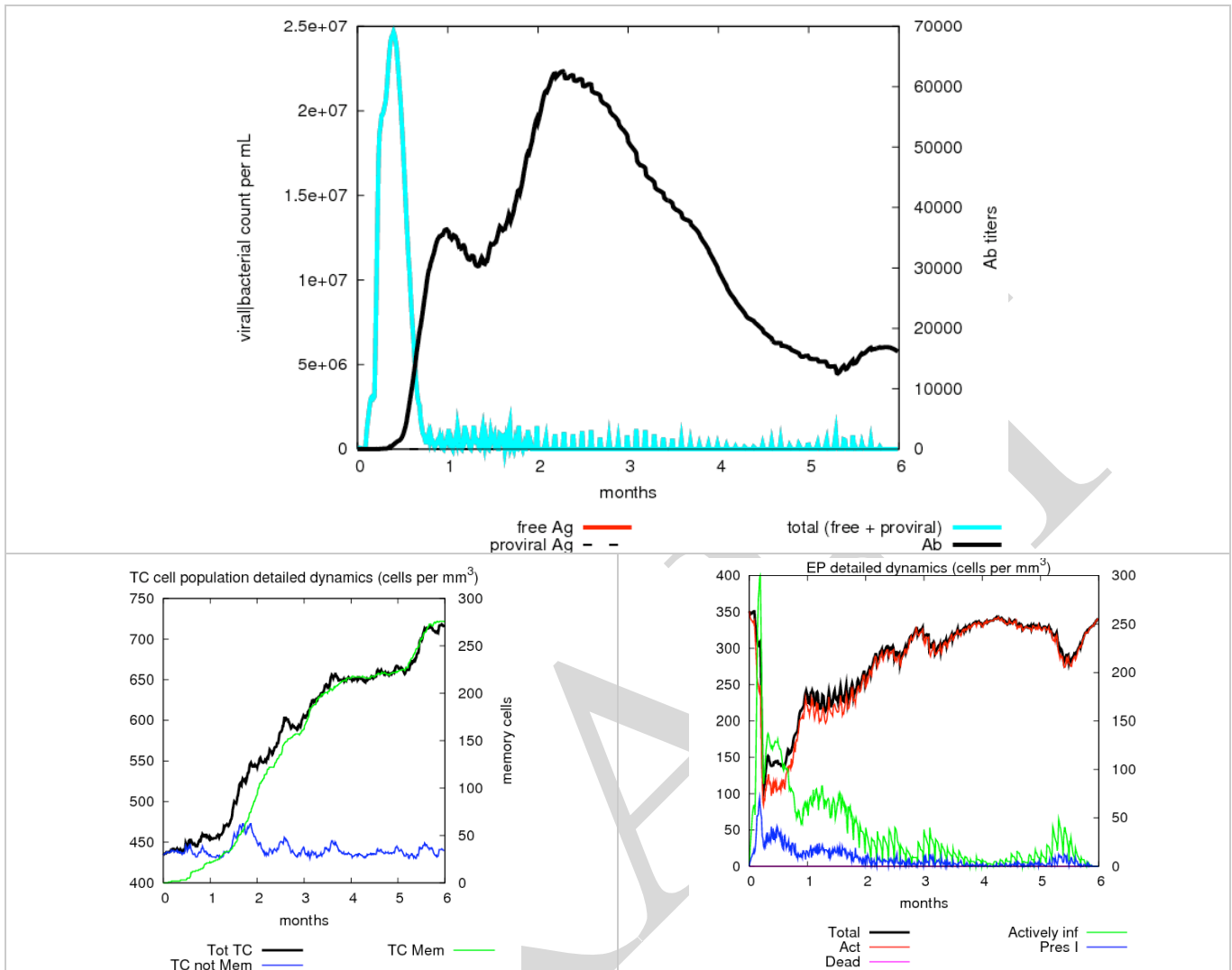
<0.5> rep rate

<100> Num Ag to inj

...  
<1234> epitope

...  
<50> NmaxVirus

...  
Figure 7 shows quite a few things. First of all in the panel at the top we observe an acute phase followed by a chronic phase due to the appearance of virus specific antibodies. Spikes in the chronic phase reveal cells that blow releasing their viral content. The bottom-left panel shows the clonal expansion of cytotoxic cells whereas bottom-right panel shows the total number of epithelial cells, the number of them that are presenting the viral peptides in the MHC class I context and those that are actively infected. Note that the cytotoxic response (but also the humoral response) contributes to ameliorate the condition of the simulated patient reducing the viral burden in the epithelial compartment.



**Figure 7** Top panel shows the acute phase followed by a chronic phase due to the appearance of virus specific antibodies. Spikes in the chronic phase reveal cells that blow releasing their viral content. The bottom-left panel shows the clonal expansion of cytotoxic cells whereas bottom-right panel shows the total number of epithelial cells, the number of them that are presenting the viral peptides in the MHC class I context and those that are actively infected.

## HIV infection

The simulation of the HIV-1 infection is much more complicated than the simulation of a generic epithelial infecting virus. By configuring and compiling the code as specified in the documentation one can obtain a customized version of C-ImmSim that simulates the HIV infection.

This accounts for a number of specific issues that are simply too many to be described here. The interested reader can have a look at [F. Castiglione, F. Poccia, G. D'Offizi and M. Bernaschi. Mutation, fitness, viral diversity and predictive markers of disease progression in a computational model of HIV-1 infection. *AIDS Research and Human Retrovirus*, 20(12) 1316-1325 (2004)] for details.

What is relevant here is the fact that the HIV-1 dynamics is quite difficult to be reproduced by a mathematical/computational model due to the two time scales observed in the disease progress: while the acute phase lasts for about two months, the silent or chronic phase can last for years (7-10 median time). The AIDS phase that follows the silent phase lasts on average 2-3 years. The difficulty is given by the fact that the HIV evolves by mutating at a high rate.

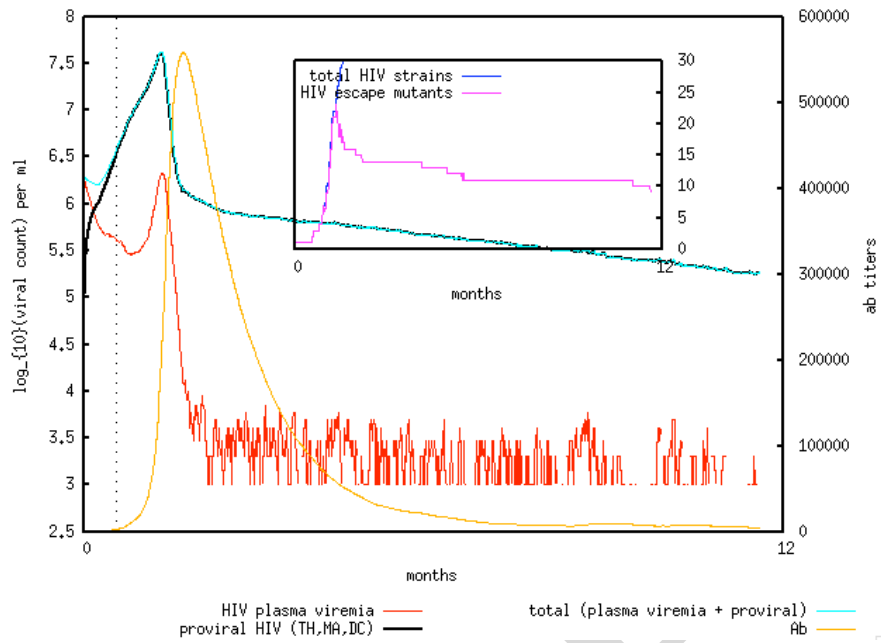
To show that C-ImmSim reproduces well the phases of disease progress (in untreated individuals) we perform a simulation by using the customized version (i.e., reconfigured and recompiled) and its input data file that contains, on top of the previous parameters, also the parameters relative to the HIV behavior.

For HIV simulation we use a different datafile called datHIV12. Since we want to make a small simulation do the following:

1. change the number of microliters (simulation space)
2. and the number of simulation steps.

```
...
<9> minmatch
<1> #microL
----- Number of time steps-----
<10950> num_step /* a year is 1095 */
...
```

Figure 8 (use the gnuplot scripts in the `../plt/Hiv/` directory) shows only the first part of the simulation where the acute phase is visible before the chronic or silent phase. Note that while the chronic phase seems to be the same as those obtained simulated a generic virus, it is not, because while battling against the immune system, the subtle HIV perform two relevant tasks: 1) mutates at a high rate and 2) impairs T helper lymphocytes production. While the former impede the immune control of the virus, the latter brings the host to the immune deficient state. Little or no T helper cells imply that an opportunistic infection will soon spread undeterred. With C-ImmSim also an opportunistic infection (like a bacterial infection of MTB) can be simulated by injecting a bacterium during the lifetime of the simulated individual (not shown).



**Figure 8** The plot shows only the first part of the simulation where the acute phase is visible before the chronic or silent phase. Note that while the chronic phase seems to be the same as those obtained simulated a generic virus, it is not, because while battling against the immune system, the HIV mutates at a high rate (see inset plot) and impairs T helper lymphocytes production (not shown).