Available online at www.sciencedirect.com
ScienceDirect
Journal of Computational and Applied Mathematics 205 (2007) 687–695
JOURNAL OF COMPUTATIONAL AND APPLIED MATHEMATICS
www.elsevier.com/locate/cam

How to deal with potentially huge dimensional state space: The meta-dynamics approach—application to a model of the co-evolution of bacterio-phage populations

Marc Baguelin[a,*], Jacques LeFèvre[b,c], Jean-Pierre Richard[d]

[a] *Cambridge Infectious Disease Consortium, Department of Veterinary Medicine, University of Cambridge, Madingley Rd., Cambridge, CB3 0ES, UK*
[b] *MIM Centre, City University, London, UK*
[c] *IDEA.SIM LTD, Harrow, London, UK*
[d] *LAGIS, Ecole Centrale de Lille, Cité Scientifique, 59651 Villeneuve d'Ascq Cedex, France*

Received 1 August 2005

Abstract

In many problems coming from the "complexity sciences", the presence of high-dimensional state spaces and non-linear equations renders traditional mathematical objects useless. To deal with these issues, several approaches have recently been studied, most of them presenting the particularity of splitting the dynamical events in two categories; an upper level in which the events describe how the structure of the system changes (the meta-dynamics) and a lower level with the events describing the evolution of the elements in the structure (the dynamics). The state space is then not defined in extenso but unfolds or contracts during the evolution of the system. In term of simulation, this view allows us to deal with only a small part of the state space at each step avoiding the time and memory limitations.

In this paper, we develop the implementation of an example of co-evolving population using a recently published unified mathematical formalism of the concept of meta-dynamics, based on an extension of Kalman's definition of dynamical systems. This general framework describes how to combine the two levels of dynamics, in order to allow the upper level (the meta-dynamics) to be able to modify the lower one (the dynamics). We explain here in detail how to use this formal approach in terms of simulation and algorithms. It allows us to present simulation results combining both changes of the structure of the system and changes at a lower population level.

MSC: 37N25; 92D15; 37M05

Keywords: Dynamical systems; Adaptive systems; Biological systems

1. Introduction

We develop here a computational methodology and describe simulations carried out on previously published work on meta-dynamics. For further details on the formalisation of that approach, one can refer to [2,7].

* Corresponding author. Tel.: +44 1223 336676.
E-mail address: mb556@cam.ac.uk (M. Baguelin).

0377-0427/$ - see front matter © 2006 Elsevier B.V. All rights reserved.
doi:10.1016/j.cam.2006.03.036

The concept of dynamical systems (DSs) is based on the assertion which states that systems can be described by picking up at some defined instants of time one element from a set—the so-called state space. The notion becomes extremely powerful when this state space can be reduced to a "simple" mathematical set ($\mathbb{R}^n$ for example with n sufficiently small) and when relatively simple laws of evolution can be written. In this paper, we are interested in the case of a huge state space with evolution laws not simple enough to allow us to tackle the high dimension of the state space by breaking it in smaller pieces (when the laws are very simple, presenting for example a high degree of linearity, it is often possible to divide the problem into smaller problems making it possible to be solved) or to lump or contract the equations of the system [4,13]. These systems, presenting an intrication of potentially huge state space and non-linear evolution laws are known as "complex" systems. Examples of these systems are numerous and can be found in many fields, among the more typical ones are: the supply of a big city such as New York in a moving economical landscape; modelling the brain where new connections between neurons appear and disappear continuously; and the colonization of an environment by a group of species which evolve to adapt to it [3].

Starting with an example of complex system from biology in Section 2, we then describe the characteristics of a general formalism to model it in Section 3. This leads us to review in Section 4 necessary considerations for anyone wanting to compute models defined with the metadynamical approach. Finally, in Section 5, numerical experiments from the biological example are implemented and results from simulations presented.

2. An example from biology: the co-evolution of bacteria and phages

The problem of the dynamics of viruses attacking bacteria populations (such viruses are called bacteriophages or phages) is of high economical interest especially in the food industry. Population models which have been designed do not consider the evolutionary aspects of the process [6], though one of the main characteristics of bacterial and phages populations are their high variabilities. On the other hand, some models have been set considering only these evolutionary aspects [12], though the interactions between the populations are known to play a major part in the evolutionary directions taken by these populations. Our model addresses both the population and evolutionary levels.

2.1. Differences with Mosekilde pre-existing model

To design our model we have used as a starting point a model from the literature [6]. This model describes the dynamical behaviours of n species of bacteria $\{B_i\}_{0\leqslant i\leqslant n}$ and phages $\{P_j\}_{0\leqslant j\leqslant n}$. The number of couple bacteria-phages is fixed and the interaction between phages and bacteria is described by ω_{ij}, the probability that a phage of P_j infects a bacterium of B_i which is also fixed (to render the specificity of a couple, ω_{ij} is high when $i = j$ and low when $i \neq j$). This system (once the number n of couples is chosen) is fixed and does not evolve.

On the contrary, our system can evolve, and to allow for this we have introduced two important changes to the Mosekilde model:

(1) To each strain of bacteria i and phage j is attached a bit string of length n_c, coding for the characteristics of the strain. ω_{ij} then depends on the similarity between the bit string of phage j and the bit string of the bacteria i.
(2) The numbers of phage strains (n_p) and of bacteria strains (n_b) no longer need to be equal. Furthermore, these numbers can vary during the simulation.

These two major differences represent features of the populations that can evolve during the simulation of the model. For this, rules on top of the dynamics are defined for mutation (apparition of a population with new characteristics) or disparition of a weak population. These rules are called meta-dynamics. In terms of simulation, the presence of these rules (mutation and disparition) means that new equations appear or disappear during a simulation. The dimension of the system can then vary.

2.2. Equations of the system

Let us consider the following system

$$\{S, \{B_i\}_{0\leqslant i\leqslant n_b}, \{P_j, I_{1,j}, I_{2,j}, I_{3,j}\}_{0\leqslant j\leqslant n_p}\}, \tag{1}$$

where S is the concentration of nutrient; B_i are the different bacteria strains with $0 \leqslant i \leqslant n_\mathrm{b}$; P_j are the different phages strains; and $I_{k,j}$ are the number of the bacteria at stage k of infection by phage j with $0 \leqslant j \leqslant n_\mathrm{p}$.

The equations of the systems are:

$$\begin{cases} \dfrac{\mathrm{d}B_i}{\mathrm{d}t} = \dfrac{vSB_i}{\kappa + S} - B_i \sum\limits_{j=1}^{n_\mathrm{p}} \alpha\omega_{ij} P_j - \rho B_i, \\ \dfrac{\mathrm{d}I_{1,j}}{\mathrm{d}t} = P_j \sum\limits_{i=1}^{n_\mathrm{b}} \alpha\omega_{ij} B_i - 3\dfrac{I_{1,j}}{\tau} - \rho I_{1,j}, \\ \dfrac{\mathrm{d}I_{2,j}}{\mathrm{d}t} = \dfrac{3}{\tau}(I_{1,j} - I_{2,j}) - \rho I_{2,j}, \\ \dfrac{\mathrm{d}I_{3,j}}{\mathrm{d}t} = \dfrac{3}{\tau}(I_{2,j} - I_{3,j}) - \rho I_{3,j}, \\ \dfrac{\mathrm{d}P_j}{\mathrm{d}t} = 3\dfrac{\beta I_{3,j}}{\tau} - P_j \left(\sum\limits_{i=1}^{n_\mathrm{b}} \alpha B_i + \sum\limits_{j=1}^{n_\mathrm{p}} \sum\limits_{k=1}^{3} \alpha I_{k,j} \right) - \rho P_j, \\ \dfrac{\mathrm{d}S}{\mathrm{d}t} = \rho(\sigma - S) - \sum\limits_{i=1}^{n_\mathrm{b}} \dfrac{v\gamma S B_i}{\kappa + S}, \end{cases} \tag{2}$$

where $0 \leqslant i \leqslant n_\mathrm{b}$, $0 \leqslant j \leqslant n_\mathrm{p}$; ρ is the rate of dilution ($\rho = 0.0045\,\mathrm{min}^{-1}$); κ and v are the saturation term and the growth from the Monod equation, respectively ($\kappa = 10\,\mu\mathrm{g\,ml}^{-1}$, $v = 0.024\,\mathrm{min}^{-1}$); α is the theoretical adsorption constant depending on phage and bacterium size ($\alpha = 10^{-9}\,\mathrm{ml\,min}^{-1}$); τ is a time constant ($\tau = 30\,\mathrm{min}$); β is the number of copies of phages j released during the burst of the infected bacterial cell ($\beta = 100$); σ is the continuous supply of substrate ($\sigma = 10\,\mu\mathrm{g\,ml}^{-1}$); γ is the amount of nutrient consumed in each cellular division ($\gamma = 0.01\,\mathrm{ng}$) and finally ω_{ij} is the probability of infection of B_i by P_j which depends on the similarity between bit string s_i^b (attached to bacterium population B_i) and s_j^p (attached to phage population P_j) as follows:

$$\omega_{ij} = \left(1 - \frac{d_\mathrm{H}(s_i^\mathrm{b}, s_j^\mathrm{p})}{n_\mathrm{c}}\right)^2$$

where n_c is the size of the binary string.

On top of this set of equations, we define two meta-dynamical rules (MRs). The first MR gives the probability $p(k)$ that a population (bacteria or phage) gives birth to k mutant populations on the interval $[t - \Delta t, t[$. This is given by

$$p(k) = \mathrm{e}^{-\lambda(t)} \frac{\lambda(t)^k}{k!},$$

where $\lambda(t) = (B_i(t)/B_\mathrm{b})\mu_\mathrm{B}$ for the bacteria and $\lambda(t) = (P_i(t)/P_\mathrm{b})\mu_\mathrm{P}$ for the phages. When k is drawn new populations of size B_b or P_b appear at a one bit distance from the mother population.

The second MR states that a population under a given threshold is removed from the system (and thus the equations of this population are removed from the system).

Then, at each step Δt there are four possible commutations:

(1) the birth of a new bacterial strain: a variable $B_{n_\mathrm{b}+1}$ is added and the dimension of the system (2) increases by one;
(2) the birth of a new phagical strain: four variables are added, $P_{n_\mathrm{p}+1}$, $I_{1,n_\mathrm{p}+1}$, $I_{2,n_\mathrm{p}+1}$ and $I_{3,n_\mathrm{p}+1}$ and the dimension of the system (2) increases by four;
(3) the extinction of a bacterial strain: the concerned variable is removed and the dimension of the system (2) decreases by one;
(4) the extinction of phagical strain: variables of the concerned phage are removed and the dimension of the system (2) decreases by four.

3. Generalisation of the approach

The approach taken in this example can be made more general. We have added an upper rule (the "meta-dynamics") to a system of differential equations from population dynamics. This rule codes the introduction of novelty in the system and thus the consequential change of structure in this system.

The idea developed in [2,7] is that what has been done with equations of population dynamics can be done with any kind of DS consistent with the Kalman definition of a DS. The MR is then viewed as an automaton which gives the possibility of commuting from one DS to another, mimicking the change of structure of the system due to the appearance of novelty.

The reason Kalman's definition has been chosen as a basis of what we have called "meta-dynamical adaptive systems" (MAS), is its generality. Other definitions of the notion of a DS have been considered, such as the behavourial approach [8]. Though it would be possible to adapt our definition of MAS to the behavourial approach, the fact that this approach uses signals and not transition functions would have made the MAS definition less straightforward. It has to be stressed also that we focus with the definition of MAS on modelling aspects. The notion of output—important in Kalman's work especially for control purposes—is not developed. There would nevertheless be no theoretical obstacles to do so.

One purpose of this paper is to show that besides having an elegant mathematical framework, the MAS approach provides also an efficient setting for computing some extremely complex systems. We here develop the approach only for the co-evolution example but the generality of the formalism can extend to other complex systems. A few examples of systems under the umbrella of this formalism are:

- hybrid systems with a finite automata playing the role of meta-dynamics upon a continuous system;
- the model of the immmune system of Bagley et al. [1] which has inspired our approach of a stochastic meta-dynamics on a set of differential equations;
- the Holland formalism where the "adaptive plan" is similar to a probabilistic meta-dynamics on the discrete lower level [3];
- the recurrent neural nets from Elmann [9] with deterministic meta-dynamics on a discrete dynamical level.

4. Computational aspects of the MAS formalism

In this section, we study methods of adapting the formalism of MAS into efficient algorithms. The exact definition of MAS is given in [7]. For the purpose of clarity, the elements of the definition are briefly summarised and then followed by a discussion on their implementation at two levels: first at a general level for all complex systems and then more specifically for the simulation of our example.

4.1. A brief overview of the definition

The definition of MAS in [7] gives rigorous laws of how a certain number of mathematical objects interact. The basis of a MAS is a set of DS in the sense of Kalman [5] sharing the same time set T_0 viewed as a subset of $\mathbb{R}$. On top of this set is a function allowing us to commute from one DS to another at some instants of a "slower" time set $T_1 \subseteq T_0$. All the other axioms of the MAS definition give the way these two levels combine together to allow the MAS to evolve with time.

4.1.1. Time

The issue of time is more difficult in MAS that it is for "classical" DSs. Indeed, the same issues arise concerning the approximation of a subset of $\mathbb{R}$ by a finite automata (i.e., a program for a computer). But besides these widely discussed problems, the problem of commutation times (when the meta-dynamics make the system pass from one DS to another) has also to be taken into account. The situation is ideal when it is possible to include the commutation instants in the simulation steps. But in many cases, it cannot be done, and then other methods have to be found (see Section 4.1.4). In our example, the evolutionary mechanisms (birth of new strains, disparition of weak populations) occur much less frequently than other biological mechanisms (such as growth, dilution etc.). We thus can consider the steps used to integrate the system of differential equations sufficiently precise and consider at each of these steps what could have happened in the interval.

4.1.2. Set of state space

By definition, simulations of complex systems involve high-dimensional spaces. But very often, the use of flexible data structures can allow the representation of possible spaces in a very economical way. A data structure often used to do this is a chained list. For example, in the case of genetic evolution of populations of individuals with new characteristics appearing and disappearing, an efficient way of coding the population is to use a chained list of elements coupling a gene and its size/density. Then, only the elements present at a time in the population are kept in memory. Thus, though there is a very large number of possible state spaces because of the combinatory explosion, the size of the chained lists in memory remains relatively small.

4.1.3. Set of transition functions

Defining the set of transition functions is not problematic if the set of state spaces is cleverly defined. In the case of a chained list state space, the functions have just to be designed for it. It is usually a bit more complicated than for static objects (as for arrays or scalars), but can be handled safely with a bit of care. In our simulation, the functions—written in C—were designed to work with any possible lists such that, even if the state space changes, the function coding for the transitions remains the same. The arguments of these functions are pointers to the area of the memory where the appropriate lists are kept. The function is then able to deal with the variety of forms the data can take.

4.1.4. The meta-dynamical rule

The definition of the MR is straightforward when time is discrete. On the other hand, difficulties arise when time is continuous. Discretisation of time then has to be made in such a way that it does not "forget" instants when the MR is active. Mechanisms have thus to be implemented to detect commutations between time steps. For stochastic MR, the density of commutation has to be replaced by finite probabilities on the interval between steps. This is the approach we have followed in our model. When the step is sufficiently small, the "continuity" of the meta-dynamics is well simulated.

4.2. Processing of the generated data

Because of the potential high dimension of the state space explorated, classical methods of recording data from the simulation and process have to be adapted. Generally, it is impossible because of the high number of elements they contain to save all data coming from the simulations of complex systems. For the simulation of our systems, we created a "report language" to track periodically (after a number of simulation steps defined by the user) the data we wanted to record. In the context of population modelling it could be, for example a specific population, or all the individuals at a given distance (distance here refers to the distance in the characteristic space, that is, how close or far they are in term of evolution of their characters) etc. It is also very often useful to use "macroscopic" variables such as entropy (related to the dispersion of the data) or the positions of centres of clusters of data. The use of macroscopic variables is detailed for the simulations given in Section 5.1.

4.3. Preliminary conclusions on computing meta-dynamics

Because of the generality of the meta-dynamical approach and the broad range of systems covered, it is impossible to cover in a few pages all the problems which can arise, as it would be impossible for DSs. We have nevertheless stressed two important points on how the two time sets have to be coupled; in particular, in the case of continuous meta-dynamical time sets and of the use of flexible data structures that give its power to the meta-dynamical approach. The use of flexible data structures such as chained lists allows us to replace the potentially huge dimensional state space in favour of a dynamical state space which changes with time, unfolding or contracting during simulation.

5. Simulations of the biological example

We now give in this section some simulation results of the example of Section 2 by discussing in detail a numerical experiment.

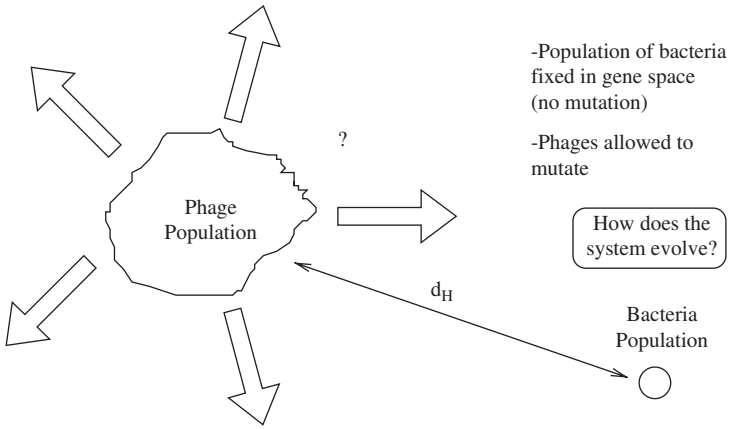

Fig. 1. Description of the numerical experiment, d_H is the original Hamming distance between the bit string of the bacteria population and the bit string of the initial phage population.

5.1. Presentation of the experiment

We have taken two populations of bacteria and phages, and set the mutation rates such that $\mu_B = 0$. That means that the characteristics of the bacteria population cannot change. On the contrary phages are allowed to mutate (because there is no ambiguity $\mu = \mu_P$ will be used from now on to refer to the rate of mutation of the phage populations). This is to simulate what could happen when $\mu_P \gg \mu_B$, when virus populations are much more variable than their hosts. We can see how the concentrations (dynamical point of view) and the composition in term of genotype (meta-dynamical point of view) evolve (Fig. 1).

5.2. Graphical output

It is interesting to see the trajectory of the cloud of phages in gene space (by this, we mean the different populations weighed by their concentrations). For visualising them we chose to plot two values: the trajectory of the centroides (the "gravity centres") and their entropies (a measure of their dispersion). The centroide c of phage population $\{P_j\}$, is the vector $c \in [0, 1]^{n_c}$ (n_c is the size of the binary string coding our genes) given by

$$c_k = \frac{1}{P} \sum_{j=0}^{n_p} P_j s_j^k,$$

where c_k is the kth component from c; s_j^k is the kth one from s_j, the string coding for the behaviour of the population P_j and $P = \sum_{j=0}^{n_p} P_j$.

The entropy is very classically given by

$$eP = -\sum_{j=0}^{n_p} \frac{P_j}{P} \log \frac{P_j}{P}.$$

Storing the trajectory of the centroides of phage populations during our simulation, we are then able to plot them using a method from statistical analysis called principal component analysis (PCA) [11]. This method allows us to find the best projection from $[0, 1]_c^n$ to $[0, 1]^2$ ("best" here means we achieve maximal conservation of distances inside the set of points).

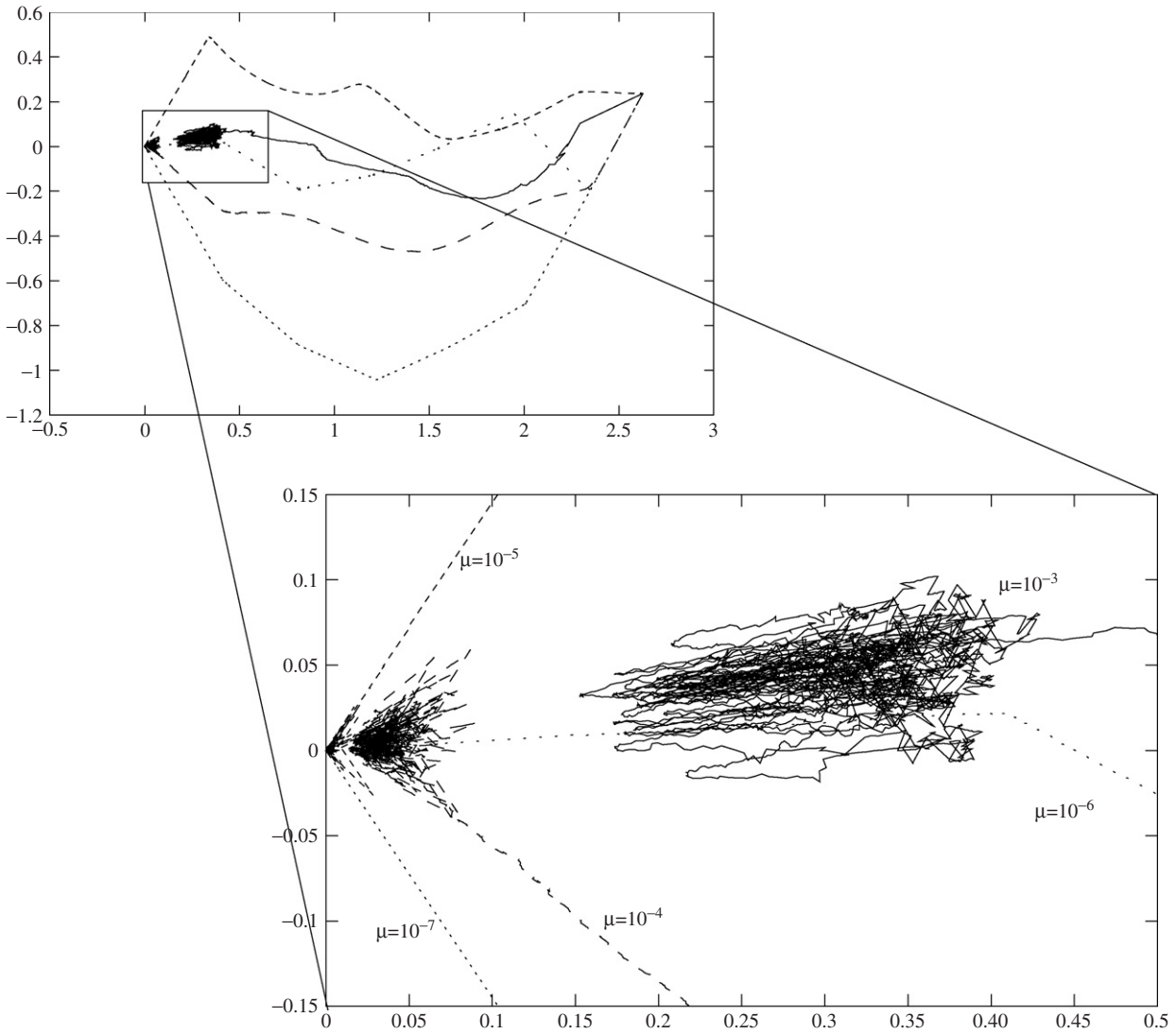

Fig. 2. Plot of the centroïde trajectories of phage populations from $[0, 1]^{10}$ vs $\mathbb{R}^2$ for five different values of μ. From $\mu = 10^{-5}$, we can observe the apparition of a quasi-species with different phenotypes present.

5.3. Results of simulation

In Fig. 2, we have plotted five different trajectories for five different values of μ. We observe, independently of μ, a similar behaviour. That is, though the populations "choose" very different paths, eventually they all reach the zone in the gene space corresponding to optimal predation.

A look at the evolution of entropy vs time (Fig. 3) gives us an idea of the mechanisms leading these experiments. That is, after a period of expansion with strong increase of entropy, the cloud contracts in a favourable area in a state of low entropy. It reaches an oscillating regime where a small number of strains co-exist. These are the best fitted phages and some at a small Hamming distance from the best fitted phages (the number of mutants and the diameter of the set of strains depend on the mutation rate μ). This organisation in a few mutant genotypes closed in a small area of the gene space make us think of a quasi-species-like organisation [10]. The quasi-species model of Eigen can thus be viewed as the asymptotical result of the evolution of our populations. This quasi-species-like organisation is a well accepted way of describing actual virus populations. Our numerical experiment thus provides a plausible scenario on how a virus

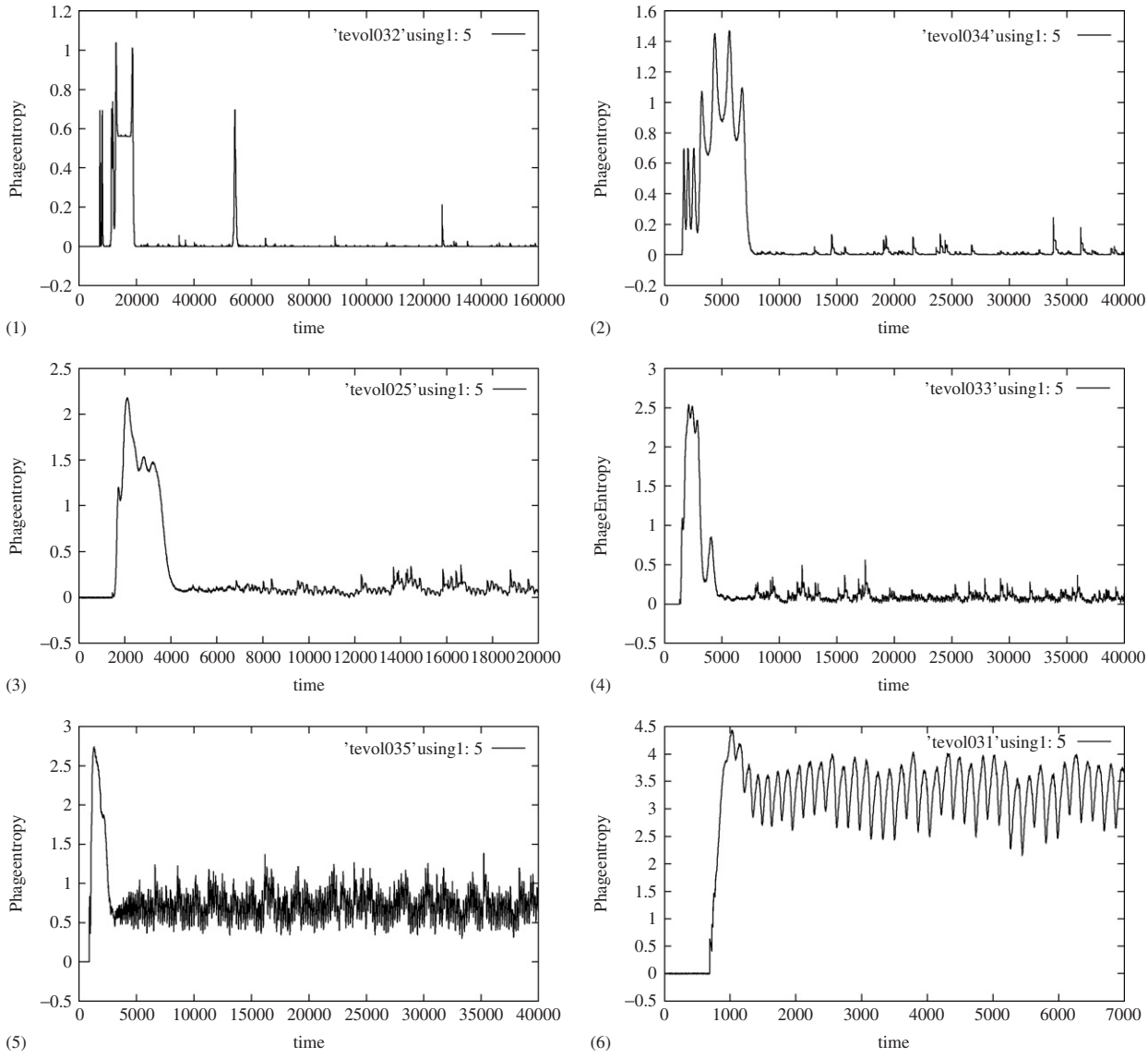

Fig. 3. Plot of the entropy of the phage population for six simulations with different values of μ and/or of the seed of the random generator: (1) $\mu = 10^{-7}$; (2) $\mu = 10^{-6}$; (3) $\mu = 10^{-5}$; (4) $\mu = 10^{-5}$; (5) $\mu = 10^{-4}$ and (6) $\mu = 10^{-3}$.

adapt to a new host. Though the "genetics" used here to represent the bacteria/phage interaction is very simple, the use of such adaptive models can be of great help to understand the process of virus adaptation.

6. Conclusions

At the level of modelling complex adaptive systems, this work demonstrates the feasibility and the efficiency of the meta-dynamical approach. Because the formalism defining meta-dynamically adaptive systems embraces a broad range of complex adaptive systems, what has been done with populations of bacteria and phages in co-evolution can be translated to many other systems.

From the biological perspective, the implementation of our example exhibits mechanisms of high importance in understanding the adaptation of viruses to their host. This model should provide clues to undertake actual experiments.

References

[1] R.J. Bagley, J.D. Farmer, S.A. Kauffman, N.H. Packard, A.S. Perelson, I.M. Stadnyk, Modeling adaptive biological systems, BioSystems 23 (1989) 113–138.
[2] M. Baguelin, Modélisation de systèmes complexes par métadynamiques: Application à la modélisation de population en coévolution, Ph.D. Thesis, Université des Sciences et Technologies de Lille, 2003.
[3] J.H. Holland, Adaptation in Natural and Artificial Systems, The University of Michigan Press, Ann Arbor, MI.
[4] M. Ikeda, D.D. Siljak, Overlapping decompositions, expansions and contractions of dynamical systems, Large Scale Systems 1 (1980) 29–38.
[5] R.E. Kalman, P.L. Falb, M.A. Arbib, Topics in Mathematical System Theory, International Series in Pure and Applied Mathematics, McGraw-Hill, New York, 1969.
[6] E. Mosekilde, H. Strandorf, J.S. Thomsen, G. Baier, Topics in Nonlinear Dynamics, World Scientific, Singapore, 1996.
[7] E. Moulay, M. Baguelin, Meta-dynamical adaptive systems and their applications to a fractal algorithm and a biological model, Physica D, 2005.
[8] J.W. Polderman, J.C. Willems, Introduction to the Mathematical Systems Theory. A Behavioral Approach, Texts in Applied Mathematics, vol. 26, Springer, NewYork, 1998.
[9] J.C. Principe, N.R. Euliano, W.C. Lefebvre, Neural and Adaptive Systems, Fundamentals Through Simulations, Wiley, New York, 1999.
[10] G.W. Rowe, Theoretical Models in Biology, The Origin of Life, The Immune System and the Brain, Oxford Science Publications, Oxford, UK.
[11] G. Saporta, Probabilités, Analyse des données et Statistique, Editions TECHNIP, Paris, France.
[12] R.S. Solé, R. Ferrer, I. González-Garcia, J. Quer, E. Domingo, Red queen dynamics, competition and critical points in a model of RNA virus quasispecies, J. Theoret. Biol. 198 (1999) 47–59.
[13] J. Tóth, G. Li, H. Rabitz, A.S. Tomlin, The effect of lumping and expanding of kinetic differential equations, SIAM J. Appl. Math. 57 (1997) 1531–1556.