Systems Biology and Advanced Computing

Alberto Maria Bersani
Dept. Mathematical Methods and Models, University "La Sapienza",
via A. Scarpa 16, 00161 Roma, Italy
bersani@dmmm.uniroma1.it
Piero Lanucara, Marco Rorro, Vittorio Ruggiero
CASPUR,
via dei Tizii 6, 00185 Roma, Italy
lanucara@caspur.it, m.rorro@caspur.it, ruggiero@caspur.it

Abstract

In this paper we show some applications of Advanced and Parallel Computing to the study of mathematical models in Systems Biology and in particular of the networks of biochemical reactions occurring inside a cell. Due to their high complexity, the numerical study of these systems must be approached by means of sophisticated Advanced Computing tools. Two examples are shown: in a deterministic framework, the application of Optimal Control techniques to the study of the effects of a drug on a cell; in the stochastic framework, the adaptation of the Gillespie algorithm to the so called total quasi steady-state approximation (tQSSA).

Keywords: Michaelis-Menten Kinetics, Quasi Steady-State Approximation, Optimal Control, Gillespie Algorithm, Chemical Master Equation.

1. Introduction

Aim of Systems Biology is to integrate biological data as an attempt to understand how biological systems function. Much of Systems Biology is within the realm of molecular biology and physiology, but it demands tools from bioinformatics and experimental data mining, mathematical modelling, proteomics and clinical sciences.

Due to the high complexity of these systems, their numerical treatment implies the usage of efficient numerical algorithms and schemes and high performance hardware and software: in other words, Advanced Computing tools.

In this paper we will show two examples, related to the mathematical models of the intracellular signal transduction networks, i.e., of the biochemical reactions induced by the cell as a response to external stimuli, like hormones, growth factors, cytokines, heat shocks etc. [1]

A prominent transduction mechanism is the phosphorylation by specific enzymes called kinases, and the dephosphorylation by phosphatases. These reactions are organized in complex networks whose behavior is determined by the activity and the interactions of each component.

The model of biochemical reactions was set forth by Michaelis and Menten in 1913 [2], and further developed by Briggs and Haldane in 1925 [3]. This formulation considers a reaction where a substrate S binds an enzyme E reversibly to form a complex C. The complex can then decay irreversibly to a product P and the enzyme, which is then free to bind another molecule of the substrate. This process is summarized in the scheme

$$E + S \underset{d}{\overset{a}{\rightleftarrows}} C \xrightarrow{k} E + P, \tag{1}$$

where a, d and k are kinetic parameters (supposed constant) associated with the reaction rates.

The formulation leads to an ODE for each involved complex and substrate. We refer to this as the full system. For (1) the equations are

$$\begin{aligned} \tfrac{dS}{dt} &= -a(E_T - C)S + d\,C, \\ \tfrac{dC}{dt} &= a(E_T - C)S - (d + k)C, \end{aligned} \tag{2}$$

with the initial conditions

$$S(0) = S_T, \quad C(0) = 0, \tag{3}$$

and the conservation laws

$$E + C = E_T, \quad S + C + P = S_T \ . \tag{4}$$

The initial conditions give the concentrations of S and C at the beginning of the reaction, and their time development is described by the ODEs, while E and P are linked to S and C through the conservation laws. Here E_T is the total enzyme concentration assumed to be free at time $t = 0$. Also the total substrate concentration, S_T, is free at $t = 0$. This is the so-called Michaelis-Menten (MM) kinetics [2, 4].

Assuming that the complex concentration is approximately constant after a short transient phase leads to the usual Michaelis-Menten (MM) approximation, or *standard quasi steady-state approximation* (standard QSSA, sQSSA). It leads to an ODE for each substrate while the complexes are assumed to be in a quasi-steady state (i.e., $\frac{dC}{dt} \approx 0$):

$$\frac{dS}{dt} \approx -kC \approx -\frac{V_{max} S}{K_M + S}, \quad S(0) = S_T, \tag{5}$$

$$V_{max} = k\,E_T, \quad K_M = \frac{d+k}{a}\ . \tag{6}$$

The advantage of a quasi steady-state approximation is that it reduces the dimensionality of the system, passing from two equations (*full system*) to one (*MM approximation or sQSSA*) and thus speeds up numerical simulations greatly, especially for large networks as found *in vivo*. Moreover, the kinetic constants in (1) are usually not known, whereas finding the kinetic parameters for the MM approximation is a standard *in vitro* procedure in biochemistry. See e.g. [4] for a general introduction to this approach. We stress here that this is an approximation to the full system, and that it is only valid when the enzyme concentration is much lower than either the substrate concentration or the Michaelis constant K_M, i.e., $E_T \ll S_T + K_M$ [5]. This condition is usually fulfilled for *in vitro* experiments, but often breaks down *in vivo*. We refer to [6] for a nice, general review of the kinetics and approximations of (1).

However, as mentioned above, to simulate physiologically realistic *in vivo* scenarios, one faces the problem that the MM approximation is no longer valid [7]. Hence, even though the kinetic constants such as K_M are identical *in vivo* and *in vitro*, they need to be implemented in an approximation which is valid for the system under investigation.

A recent approach to resolve this problem is that of the total quasi-steady state assumption (tQSSA). It was introduced by Borghans et al. [8] and refined by Tzafriri [9] for isolated reactions.

This approximation holds for a much larger region of parameter space, and is in fact always roughly valid [9]. Importantly, the tQSSA coincides with the sQSSA when the latter is expected to hold, i.e., at low enzyme concentrations.

The tQSSA [8, 9] arises by introducing the total substrate

$$\bar{S} = S + C, \tag{7}$$

and assuming that the complex is in a quasi-steady state as for the sQSSA. For (1) it gives [9]

$$\frac{d\,\bar{S}}{dt} \approx \ -k\,C_-(\bar{S}), \quad \bar{S}(0) = S_T, \tag{8}$$

where

$$C_-(\bar{S}) = \frac{(E_T + K_M + \bar{S}) - \sqrt{(E_T + K_M + \bar{S})^2 - 4E_T\bar{S}}}{2}. \tag{9}$$

Numerical integration of (8) easily gives the time behavior of $\bar{S}, C$ (by (9)) and S (by the relation $S = \bar{S} - C$).

Tzafriri [9] showed that the tQSSA (8) is valid whenever

$$\epsilon_{Tz} := \frac{K}{2S_T}\left(\frac{E_T + K_M + S_T}{\sqrt{(E_T + K_M + S_T)^2 - 4E_T S_T}} - 1\right) \ll 1, \quad \text{where} \quad K = \frac{k}{a}, \tag{10}$$

and that this is always roughly valid in the sense that

$$\epsilon_{Tz} \leq \frac{K}{4K_M} \leq \frac{1}{4}. \tag{11}$$

This means that for *any* combination of parameters and initial conditions (8) is a decent approximation to the full system (2). The parameter K is known as the Van Slyke-Cullen constant.
This paper is organized as follows.
In Section 2 we will show an example of application of Optimal Control techniques (via Hamilton-Jacobi equations) to the study of the effects of a drug on the single Michaelis-Menten reaction, in order to control the degradation of the product P, supposed toxic.
In Section 3 we will show some more complex reaction networks, in a stochastic framework, due to the presence of a small number of molecules, and adapt the so called Gillespie Algorithm, or Stochastic Simulation Algorithm (SSA), to the tQSSA. The examples are taken from [10].

2. Effects of a drug on cellular enzymes with degradation. Optimal Control and the Hamilton-Jacobi equation

When we consider reaction (1) we can introduce a term which describes the biochemically relevant phenomenon of degradation, or death, of some enzymes. The degradation can be induced or accelerated by some drugs, which can act directly on a specific enzyme, which can be considered toxic for the cell. Our aim is to control the degradation rate of P, taking into account practical limitations, like the toxicity and/or the costs of the drug.
In the framework of the Dynamic Programming approach we can exhibit a mathematical formulation of the above described problem.
Introducing in (1) a suitable control $k_\alpha(t)$ we obtain the supplementary equation

$$\frac{dP}{dt} = k\ C(t) - k_\alpha P(t)\ , \tag{12}$$

where the term $k_\alpha P(t)$ represents the degradation of P.
Let us impose that the dynamical system evolves inside a subset Ω of $I\!R^3$

$$\Omega = \{(S, C, P) \in I\!R^3 | S \geq 0\ ,\ C \geq 0\ ,\ P \geq 0\}\ . \tag{13}$$

Interpreting the function $k_\alpha(t)$ as a control, we want to determine the admissible controls k_α such that

$$(S(t), C(t), P(t)) \in \Omega\ , \tag{14}$$

$$\lim_{t\to+\infty} P(t) = 0 \tag{15}$$

and $k_\alpha \in [0, K]$ for some $K > 0$.
The upper bound K can be interpreted as the need of modeling the presence of cost(s) and/or toxicity of the drug.
Let us call U_{ad} such a set of controls.
We rewrite system (2), (12) in the vector form

$$\frac{dy}{dt} = g(y(t), k_\alpha(t)) \tag{16}$$

where

$$y(t) = (S(t), C(t), P(t))^T \in \mathbb{R}^3 \tag{17}$$

$$y(0) = (S_T, 0, 0)^T =: x \in \mathbb{R}^3 \tag{18}$$

and

$$\begin{aligned} &g(y(t), k_\alpha(t)) = (-a(E_T - C(t))S(t) + d\, C(t)\ , \\ &a(E_T - C(t))S(t) - (d + k)C(t)\ ,\ k\, C(t) - k_\alpha P(t))^T\ . \end{aligned} \tag{19}$$

For any "admissible control" we define a "cost" functional, depending on the choice of the initial condition $y(0) = x$, as

$$J_x(k_\alpha(\cdot)) = \int_0^{+\infty} l(y_x(t), k_\alpha(t)) \cdot e^{\int_0^t c((y(s),k_\alpha)ds} dt\ \geq\ 0 \tag{20}$$

where l is an *ad hoc* function and c is a nonnegative continuous function on $\overline{\Omega}$.
In our first models we have decided to put $c \equiv 0$.
Putting

$$u(x) = \inf_{k_\alpha(\cdot)\in L^\infty} J_x(k_\alpha(\cdot))\ , \tag{21}$$

the problem is to determine $k_\alpha(\cdot)$ such that

$$u(x) = \min_{k_\alpha(\cdot)\in L^\infty} J_x(k_\alpha(\cdot))\ . \tag{22}$$

It is well-known [11] that $u(x)$ is a solution of the following boundary value problem:

$$H(x, u(x), \nabla u(x)) = 0 \text{ on } \Omega \tag{23}$$

$$u(x) = 0 \text{ on } \partial\Omega\ , \tag{24}$$

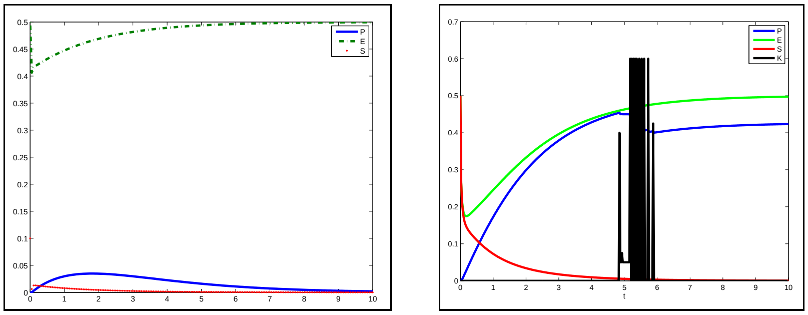

Figure 1: The effect of control k_α on P degradation. $K = 0.6$. *Left:* $l(y_x, k_\alpha) = P$; $E_T = 0.5; S_T = 0.1$. In this case, k_α (not shown) immediately reaches the maximum value K and P can be completely degraded. *Right:* $l(y_x, k_\alpha) = P^2 + k_\alpha$; $E_T = 0.5; S_T = 0.5$. Due to the presence of costs related both to P and k_α, the control chooses to degrade only partially P.

where the Hamiltonian function H is defined as

$$H(x, u(x), \nabla u(x)) = \min_{k_\alpha \in U_{ad}} \{g(x, k_\alpha)\nabla u(x) + l(x, k_\alpha)\} \ . \tag{25}$$

As shown in Fig. 1, the choice of the function l is fundamental.

The starting point for numerical simulations was the code developed by Carlini et al. [12] for solving Hamilton-Jacobi equations in high spatial dimensions. This code is able to solve problems in 2 or 3 spatial dimension but suffers of the so–called ”curse of dimensionality” implicit in this kind of mathematical formulation.

For this reason we developed a parallel version able, at this moment, to run efficiently up to 4 or 5 spatial dimensions (that is equivalent to say 4 or 5 equations). The preliminary parallelization has been implemented using OpenMP, a standard tool designed for shared memory architectures (including new multicore machines).

OpenMP is a directive based approach to the parallelization and provides support for concurrency, synchronization, and data handling while obviating the need for explicitly setting up mutexes, condition variables, data scope, and initialization. A typical OpenMP program executes serially until it encounters the parallel directive. This directive is responsible for creating a group of threads. The core of the parallel code follows. It should be noted that the OpenMP specific part in the parallel code is very small:

```
!$omp parallel default(shared)&
!$omp do private(jj,nodo,ctrl,in,l,stimaHJ)
                do jj=1,nnodi
                  call Inv_ind(jj,n,In)
                  do l=1,m
                     nodo(l)=x(In(l)+last(l))
```

```
         enddo
         call STIMAINICTRL(stimaHJ,nodo,ctrl)
         u(jj)=stimaHJ
         uctrl(jj)=ctrl(1)
        end do
!$omp end parallel
```

Further optimizations should concern the use of distributed memory techniques and tools (MPI) in order to run efficiently on distributed memory supercomputers using a greater RAM than in a single machine.
The parallel machine used to conduct the numerical tests is a 8 processors IBM System p5 575, working at 1900 MHz clock, with a theoretical peak performance of 60 GFLOPS.

3. The total quasi steady-state approximation (tQSSA) and the stochastic framework

For this Section we refer to [10], where all the citations to the related literature can be found, together with several other examples.
Under some circumstances – for example, when some species are present in small numbers – a discrete and stochastic framework is the most appropriate framework for modeling chemical kinetics. Such a framework is provided by continuous-time, discrete-state Markov processes.
A very popular method for studying and simulating intrinsic noise is the Stochastic Simulation Algorithm (SSA) [13].
In the presence of both fast and slow reactions the quasi-steady-state approximation (QSSA) has been one such multiscale method that has recently received much attention for the purpose of speeding up the SSA.
For a recent review of the evolution of SSA see [10] and references therein.
In [10] MacNamara et al. investigated its application to the direct solution of the Chemical Master Equation (CME), which describes the evolution of the probability mass function associated with the SSA. Significantly they were able to adapt a CME-solver, based on Krylov methods, by incorporating tQSSA and thus take advantage of the multiscale nature of the systems being studied.
Rather than simulating a path through the Markov process, we can, given an initial condition $\mathbf{x}(t_0) = \mathbf{x_0}$, directly compute the probability of being in state $\mathbf{x}$ at time t, $P(\mathbf{x}; t)$. It can be shown that for each state $\mathbf{x}$, the previous description of the model implies that this probability satisfies the following discrete PDE

$$\frac{\partial P(\mathbf{x}; t)}{\partial t} = \sum_{j=1}^{M} \alpha_j(\mathbf{x} - \nu_j) P(\mathbf{x} - \nu_j; t) - P(\mathbf{x}; t) \sum_{j=1}^{M} \alpha_j(\mathbf{x}), \tag{26}$$

where the so-called *stoichiometric* vector ν_j, of the same dimension as the state vector, defines the way the state changes when a reaction occurs; if the system is in state $\mathbf{x}$ and reaction j occurs, then the system transitions to state $\mathbf{x} + \nu_j$.

Associated with each state is a set of M *propensities*, $\alpha_1(\mathbf{x}), ..., \alpha_M(\mathbf{x})$, that determine the relative change of each reaction occurring.

	Reaction	Propensity
1	$S_1 + E \longrightarrow C_1$	$c_1 \times S_1 \times E$
2	$S_1 + E \longleftarrow C_1$	$c_2 \times C_1$
3	$C_1 \longrightarrow S_2 + E$	$c_3 \times C_1$
4	$S_2 + E \longrightarrow C_2$	$c_4 \times S_2 \times E$
5	$S_2 + E \longleftarrow C_2$	$c_5 \times C_2$
6	$C_2 \longrightarrow P + E$	$c_6 \times C_2$

Table 1: Description of the dual phosphorylation enzyme kinetics scheme. Examples (a), (b) and (c) have $c = [0.2,\ 1.0,\ 0.6,\ 0.2,\ 1.0,\ 0.5]^T$ [14] with initial states $[100,\ E_I,\ 0,\ 0,\ 0,\ 0]^T$, where $E_I = 1000, 100, 10$ with a corresponding choice of $t_f = 2, 2.5, 20$. Examples (d), (e) and (f) match (a), (b) and (c), respectively, except they have different rate constants: $c = [1.0,\ 1.0,\ 0.1,\ 1.0,\ 1.0,\ 0.1]^T$.

This Chemical Master Equation may be written in an equivalent matrix-vector form so that the evolution of the probability density $\mathbf{p}(t)$ (which is a vector of probabilities $P(\mathbf{x}; t)$, indexed by the states $\mathbf{x}$) is described by a system of linear, constant coefficient, ordinary differential equations:

$$\dot{\mathbf{p}}(t) = \mathbf{A}\mathbf{p}(t),$$

where the matrix $\mathbf{A} = [\mathbf{a_{ij}}]$ is populated by the propensities and represents the *infinitesimal generator* of the Markov process, with $a_{jj} = -\sum_{i \neq j} a_{ij}$.

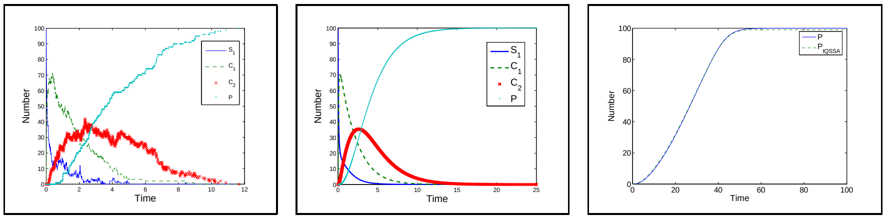

Figure 2: Dynamics of the stochastic and deterministic models of dual phosphorylation for example (b) in Table 1. *Left:* A stochastic trajectory (via the SSA). The process is absorbed at $t_f \approx 12$. *Middle:* Solution to the corresponding deterministic model. The system is just beginning to settle down by $t = 12$. *Right:* The tQSSA solution of the deterministic double phosphorylation model for example (c) in Table 1. The approximation is very good, although it slightly overestimates the population of products for very early times and slightly underestimates the population at equilibrium.

Example		runtime (s)	$\|\cdot\|_1$	$\|\cdot\|_2$	$\|\cdot\|_\infty$	n	$\mathbb{E}(P)$
(a)	**A**	7,446				4,517,885	29.4
	B	356	4E-2	8E-3	3E-3	5,151	29.7
(b)	**A**	1,414				4,517,885	28.2
	B	353	0.6	0.1	4E-2	5,151	31.8
(c)	**A**	60				270,272	31.4
	B	5	0.3	5E-2	1E-2	5,007	33.3
(d)	**A**	7,567				1,782,721	1.745
	B	144	2E-3	1E-3	6E-4	5,151	1.749
(e)	**A**	1,227				1,869,423	2.1
	B	151	9E-2	4E-2	2E-2	5,007	2.2
(f)	**A**	5				32,967	1.6
	B	1.2	6E-2	3E-2	2E-2	5,007	1.7

Table 2: Comparison of Krylov FSP (**A**) and tQSSA (**B**) for the double phosphorylation model, with examples as in Table 1. The accuracy of the tQSSA is assessed in terms of the conditional distribution for the products, P, the mean of which is recorded in the last column. For each method, n is the size of the projection used.

Given an initial distribution $\mathbf{p}(\mathbf{0})$, the solution at time t is

$$\mathbf{p}(t) = \exp(t\mathbf{A})\mathbf{p}(0), \tag{27}$$

where the exponential of a bounded operator is usually defined via a Taylor series: $\exp(t\mathbf{A}) = \mathbf{I} + \sum_{n=1}^{\infty} \frac{(t\mathbf{A})^n}{n!}$. The Finite State Projection (FSP) algorithm uses a truncated version of the full operator, which is always finite and bounded, and which provides an approximation to the behaviour of the model.

The FSP method was recently improved to a Krylov-based approach, by adapting Sidje's Expokit codes.

The Krylov FSP converts the problem of exponentiating a large sparse matrix to that of exponentiating a small, dense matrix in the Krylov subspace.

The tQSSA is the most natural aggregation to consider in the stochastic regime. In [10] MacNamara et al. analyzed the application of tQSSA to the CME-solver. Here we show only the case of the double phosphorylation mechanism (see Tables 1, 2 and Figure 2), which was studied in a deterministic framework by Pedersen et al. [15]. MacNamara et al. compared the accuracy and efficiency of two numerical methods for the solution of the CME: (A) the Krylov FSP for the full CME and (B) the tQSSA-based CME-solver. All numerical experiments used FORTRAN with the Intel 'ifort' compiler, and were conducted on an SGI Altix with 64 Itanium 2 CPUs and 120 GB of memory running the Linux operating system. However only a single processor was used. Since the true solution is not available they assessed the accuracy of the tQSSA by comparison with the Krylov FSP, with strict tolerances. By default the Krylov FSP is called with (Expokit, FSP) tolerances of $(10^{-8}, 10^{-5})$ but these bounds are pessimistic –

the actual results may be better.

References

[1] U. S. Bhalla, Models of cell signaling pathways, *Curr. Opin. Genet. Dev.* **14** (2004), 375-381.

[2] L. Michaelis & M. L. Menten, Die kinetik der invertinwirkung, *Biochem. Z.* **49** (1913), 333-369.

[3] G. E. Briggs & J. B. S. Haldane, A note on the kinetics of enzyme action, *Biochem. J.* **19** (1925), 338-339.

[4] H. Bisswanger, *Enzyme Kinetics. Principles and Methods*, Wiley-VCH, 2002.

[5] L. A. Segel & M. Slemrod, The quasi steady-state assumption: a case study in perturbation, *SIAM Rev.* **31** (1989), 446-477.

[6] S. Schnell & P. K. Maini, A century of enzyme kinetics. Reliability of the KM and vmax estimates, *Comments Theor. Biol.* **8** (2003), 169-187.

[7] M. G. Pedersen, A. M. Bersani & E. Bersani, Quasi Steady-State Approximations in Intracellular Signal Transduction – a Word of Caution, Preprint Me.Mo.Mat. n. 6/2006, accepted for publication in *J. Math. Chem.*

[8] J. Borghans, R. de Boer & L. Segel, Extending the quasi-steady state approximation by changing variables, *Bull. Math. Biol.* **58** (1996), 43-63 .

[9] A. R. Tzafriri, Michaelis-Menten kinetics at high enzyme concentrations, *Bull. Math. Biol.* **65** (2003), 1111-1129.

[10] S. MacNamara, A. M. Bersani, K. Burrage & R. B. Sidje, Stochastic chemical kinetics and the *total* quasi-steady-state assumption: application to the stochastic simulation algorithm and chemical master equation, submitted to *J. Chem. Phys.*.

[11] M. Bardi & I. Capuzzo-Dolcetta, *Optimal Control and Viscosity Solutions of Hamilton-Jacobi-Bellman Equations*, Birkhäuser, 1997.

[12] E. Carlini, M. Falcone & R. Ferretti, An efficient algorithm for Hamilton-Jacobi equations in high dimension, *Computing and Visualization in Science* **7** (2004), 15-29.

[13] D. T. Gillespie, *Markov Processes: An Introduction for Physical Scientists*, Academic Press, 1992.

[14] M. G. Pedersen, A. M. Bersani, E. Bersani & G. Cortese, The Total Quasi-Steady State Approximation for Complex Enzyme Reactions, accepted for publication in *Mathematics and Computers in Simulation.*

[15] M. G. Pedersen, A. M. Bersani & E. Bersani, The Total Quasi Steady-State Approximation for Fully Competitive Enzyme Reactions, *Bull. Math. Biol.* **69** (2005), 433-457.