MISSION-T2D

Multiscale Immune System SImulator for the Onset of Type 2 Diabetes integrating genetic, metabolic and nutritional data

Work Package 2

Deliverable 2.2

Report on deterministic and stochastic modelling

European Commission

SEVENTH FRAMEWORK PROGRAMME

Document Information

Grant Agreement	**Nº**	600803	**Acronym**	MISSION-T2D
Full Title	Multiscale Immune System SImulator for the Onset of Type 2 Diabetes integrating genetic, metabolic and nutritional data			
Project URL	http://www.mission-t2d.eu			
EU Project Officer	**Name**	Dr. Adina Ratoi		

Deliverable	**No**	2.2	**Title**	Report on deterministic and stochastic modelling
Work package	**No**	2	**Title**	Clinical data provision (genetics and aging) and gut microbiota modeling

Date of delivery	**Contractual**		**M12**		**Actual**		**M12**	
Status	**Version 1.3**				**Final**			
Nature	**Prototype**		**Report**	X	**Dissemination**		**Other**	

Dissemination level	Consortium+EU	
	Public	X

Target Group (If Public)		Society (in general)	
Specialized research communities	X	Health care enterprises	
Health care professionals		Citizens and Public Authorities	

Responsible Author	**Name**	Stefano Salvioli	**Partner**	UniBO
	Email	stefano.salvioli@unibo.it		

Version Log

Issue Date	Version	Author (Name)	Partner
23.02.2013	1.1	Gastone Castellani	UniBO
26.03.2013	1.2	Filippo Castiglione	CNR
05.03.2013	1.3	Stefano Salvioli	UniBO

Executive Summary	In this document are reported the results of the Task 2.2 “Report on deterministic and stochastic modelling”, regarding the activities of Gut Microbioma modeling. The main goal is to create a dynamical model of Gut Microbiota evolution, by taking into account the complexity of the intrinsic and extrinsic interactions of this ecological system with components such as the bacterial species, food and other environmental variables and the Immune System. After an ecological instantiation, we developed firstly a deterministic model of Gut Microbiota and simulated its temporal evolution by taking into account the interactions of species and the effect of diet. The deterministic model is then translated into a stochastic framework by using the Chemical Master Equation methodology, a novel and innovative approach, that will be used to fit experimental data distributions we have obtained in our previous deliverable.
Keywords	Large system of ordinary differential equations, stochastic simulation, Gillespie algorithm. External perturbations

Contents

1 Deliverable description

In this document are described the hypotheses, the theoretical assumptions, the methodologies and the results of the Task 2.2: “Report on deterministic and stochastic modelling”. The main focus will be on the modelling activities and we will describe the main steps we have done for the creation of such a model. The main goal is to develop a dynamical model of Gut Microbiota and characterize its response to external perturbations such as dietary changes. The model is based on a refinement of the generalized *n* species Volterra model by adding the effect of food intake. The model is hence ready to be used for fitting real metagenomic data. After this, we present a stochastic implementation of this model, that can be used for fitting the data distribution.

The numerical solution of these systems has been obtained on a dedicated server with a general purpose software developed in Python, C and Mathematica. The server is a Linux 36 core cluster with about 200 Gb of RAM hosted in the Physics and Astronomy Department of the Bologna University (Partner 3) and is appropriate to perform large scale numerical simulations.

2 Introduction

The main goal of WP 2 is to implement and manage the interaction between clinical data with the immune system simulator (ISS, WP6) both as input source and as output interpretation, including validation. The Task 2.2 has to provide the framework for the setting up for solving large system of Nonlinear Ordinary Differential Equation and their stochastic generalization. The stochastic generalization is done by the so called Chemical Master Equation (CME) that is a phenomenological set of first order differential equations describing the probability of the system to be in a discrete state, that usually indicates the number of objects (in this case the integer number of bacteria).

The CME we are using is a discrete Markov process with continuous time with the additional property of “one step process” (One step Poisson process), meaning that the possible transitions are those from the nearest- neighbour states (n-1, n, n+1).

According to these properties, the general solution of the CME is a time dependent probability distribution, while the equilibrium solution is a stationary probability distribution.

Although in some cases is possible to obtain the analytical solution for the CME, and especially for the stationary distribution, we choose to develop a general module in Python and C for the numerical solution of the CME by using the Gillespie algorithm and its generalization (the tau leaping method).

The same is true also for the system of ODE, the model we used to simulate the Gut Microbiota (GM) is a simple ecological model, based on a generalization of the Lotka-Volterra equation (the so called *n* species Volterra model). This model can admit also an analytical solution and several properties can be obtained by linear stability analysis, but we preferred to develop a module for the numerical integration of the system, even if we have developed software for the linear stability analysis by using symbolic algebra systems such as Mathematica and Python, capable to perform symbolic manipulation on the systems equations.

3 Deliverable results

3.1 Ecological theories for Gut Microbiota modeling

The main purpose of modern ecological theories is to describe and explain the within-trophic-level biodiversity. Here, with the term 'biodiversity' we denote both species richness, that is the total number of species in a defined space at a given time, and relative species abundance (RSA), which refers to their commonness or rarity. Instead, with the words 'within-trophic-level' we mean that we are going to study organisms that occupy the same position in a food chain. Thus we will not consider problems such as the trophic organization of communities, or what controls the number of trophic levels, or how biodiversity at one trophic level affects diversity on other trophic levels. The reason for this is that, while not complete, a theory of biodiversity within trophic levels would nevertheless be a major advancement because most biodiversity resides within rather than between trophic levels (i.e. there are many more species than trophic levels).

In this perspective, we can define an 'ecological community' as a group of trophically similar species that exist in the same local area and that actually or potentially compete for the same or similar resources, and a 'metacommunity' as the ensemble of all trophically similar individuals and species in a regional collection of 'local communities', in which species may not actually compete because of separation in space or time.

Modern ecological theories can be distinguished in essentially two main schools of thought: the niche assembly perspective and the dispersal one.

The physicist Heinz Pagels (1982) once observed that there seem to be two kinds of people in the world. There are those who seek deterministic order and meaning in every event, and those who believe events to be influenced, if not dominated, by random chance. This is the controversy between determinism and stochasticity that dominated the twentieth-century physics, one of whose triumphs was exactly to prove that both views of physical nature are simultaneously true and correct, but on very different spatial and temporal scales. The same kind of debate also persists for example in population genetics debates, where the question is whether most changes in gene frequencies result from random evolution or from natural selection, and similarly exists in ecology, where there are these two conflicting world views on the nature of ecological communities: the niche and the dispersal perspectives.

3.2 Niche Theory

The niche assembly perspective holds that communities are groups of interacting species whose presence or absence and even their relative abundance can be deduced from deterministic 'assembly rules' that are based on the ecological niches or functional roles of each species. Here, the concept of 'ecological niche' summarizes the interactions between species and their environment, and is thus defined by two components:

- the requirement for an organism of a given species to live in a given environment (the extent to which a limiting factor, like a resource, a predator or a parasite, influences the birth and death rate of that species);
- the impact of the species on its environment (the extent to which the growth of a population alters the limiting factor, i.e. the availability of a resource or the density of a predator or parasite).

According to this view, species coexist in interactive equilibrium and a stable co-existence among competing species is made possible by niche partitioning. The stability of the community and its resistance to perturbation derive from the adaptive equilibrium of member species, each of which has evolved to be the best competitor in its own ecological niche. Niche-assembled communities are limited-membership assemblages in which interspecific competition for limited resources and other biotic interactions determine which species are present or absent from the community. We have to under line that most proponents of niche assembly come out of a strong neo-Darwinian tradition, which focuses on the lives of interacting individuals and their fitness consequences. The concept of niche follows naturally and logically as the population level summation of the individual adaptations of organisms to their environments.

Niche theory resulted able to predict patterns of species traits and species separation on nutrient gradients similar to those observed in different studies and provided a potential explanation for the high diversity of nature, predicting that habitat heterogeneity can allow a potentially unlimited number of species to co-exist if species that are better at dealing with one environmental constraint are necessarily worse at dealing with another [34]. On the other hand, this theory is not able to predict a limit to diversity, and consequently neither to explain species relative abundance.

3.3 Dispersal and Neutral Theory

The other world view is the dispersal assembly perspective, which asserts that communities are open, non-equilibrium assemblages of species largely thrown together by chance, history, and random dispersal. Species come and go, their presence or absence is dictated by random dispersal and stochastic local extinction.

Actually we will refer to a particular class of dispersal theories, those called ‘neutral’, in which ecological communities are structured entirely by ecological drift (i.e. demographic stochasticity), random migration, and random speciation. By neutral we mean that the theory treats organisms in a trophically defined community as essentially identical in their per capita probabilities of giving birth, dying, migrating, and speciating (ecological equivalence). We have to underline that neutrality is defined at the individual level, not at the species level, thus this is a very unrestrictive and permissive definition since it does not preclude interesting biology from happening or complex ecological interactions from taking place among individuals. All that is required is that all individuals of every species obey exactly the same rules of ecological engagement. So, for example, if all individuals and species enjoy a frequency-dependent advantage in per capita birth rate when rare, this per capita advantage will be exactly the same for each and every individual of a species of equivalent abundance.

One consequence of a focus on adaptation and niche assembly has been a tendency to accept a equilibrium and a relatively static view of niches and ecological communities. This focus on individual variation in fitness, adaptation and niche, moreover, has led naturally to small-scale, short-term experimental studies of processes of competition, selection and adaptation. Proponents of dispersal assembly criticize this and typically work on much larger spatial and temporal scales, using biogeographic or paleo-ecological frames of reference, through an approach less experimental and more analytical of large-scale statistical patterns.

Thus for example, as reported in literature, data from much fossil records revealed that many pre-Holocene, full glacial, and previous interglacial plant communities are very different from modern communities. The evidence from many studies is strong that communities undergo profound compositional changes, sometimes gradual, sometimes episodic, on timescales of centuries to millennia and longer.

The fact is that species are transient, even if transit time to extinction are often of the order of millions or tens of millions of year, and furthermore in most of the cases local extinction can not be attributed to competitive exclusion. So, as suggested by Hubbell in his work, we should not concentrate on the indefinite coexistence of specie, but rather on the study of species presence-absence, persistence times, and above all species relative abundance (RSA) in communities, that can be compared with real data.

3.4 The Gut Microbiota model: biological and ecological background

The **human metagenome** is the set of the Homo sapiens genes plus the trillions of genes in the genomes of microbes that live in the human body. The microbial genome (**microbiome**) is in a dynamical relation with the human organism and helps it in carrying out crucial functions such as metabolic processes, (food absorption, short chain fatty Acid (SCFA) and vitamins production), shaping, control and **protective Immune (IS) system development,** that helped the **(co)-evolution** of human being.

With the term Metagenomics, we define the set of omics measurements aimed to quantify the composition and the interactions dynamics between the host and

the microbiome. This includes characterization at the level of DNA (metagenome), RNA (`meta'-transcriptome), protein (meta-proteome) and metabolic network (`metabolome'), both for the host and the microbiome.

Hence, H.sapiens is a **metaorganism (or super organism)** where the different microbiota present in different organs play a major physiological and pathological role. We will refer in particular to the **Gut Microbiota** (phylogenetic) and **Gut Microbiome** collectively indicated here as **GM**.

The **bidirectional cross talk** between host and GM is supported by several experimental data (pre- and pro-biotics, GM transplants, antibiotics, GM transplants results, induction of donor phenotypes in the host, including the recovery of a sick recipient) and from the **association of GM composition with pathological states,** such as Obesity, Aging. GM is sensitive to environmental stimuli (particularly to nutrition), has an high **individual specificity**, **plasticity** and is modifiable

A crucial **metagenomic quantity** is the **intersection between the host and the microbiome**, this interface is the way by which the host and the microbiota interact. This interaction is **personalized, dynamic, bidirectional and history-dependent** and is taking place in a multivariate way, by exchange of various molecules: metabolic, genetic, immunitary, etc.

The **dynamic properties of the GM** are caused by the fact that GM is a **complex ecosystem with a complex dynamics** derived by the interactions with components such as the **virome** (the set of viruses in the human body) and the **Immune System**, that can be modeled by classical predator-prey and ecological/microbial growth equations (Lotka-Volterra, Chemostat, etc).

The classical Lotka-Volterra system has been applied to model microbial community in various contexts, both with continuous and discrete time. The way to introduce the effect of diet and/or antibiotics is to introduce an external perturbation that is added to the n species Volterra model.

Formally, this model consists of autonomous, non-linear, coupled first-order ordinary differential equations.

a) Predator Prey model (Lotka Volterra model)

We report the classical model only for the sake of simplicity, and to point out that this model has been used for describing the dynamics of of GM bacteria and viruses and was capable to provide a solution consistent with the so called “Red Queen Dynamics”.

$$\begin{cases} \dfrac{dx}{dt} = \alpha \cdot x - \beta \cdot x \cdot y \\ \dfrac{dy}{dt} = \delta \cdot x \cdot y - \gamma \cdot y \end{cases}$$

In this model, x is the number of preys, y is the number of predators, while a,b,c,d are respectively: the growth rate of the preys, the rate of predation (how many preys are eaten by the predators), the reproduction rate for the predators (proportional to the number of eaten preys) and the death rate of the predators (proportional to the inverse of the lifespan).

b) N species Volterra model

A generalization to n species of the classical LV model is obtained by introducing the interaction matrix M

$$\frac{dx_i}{dt} = x_i\left(\sum_{j=1}^{n} M_{i,j} x_j - r_i\right)$$

x_i indicates the number of the I-th bacterial species and ri is the natural birth or death of the i-th specie in absence of all the other species. The sign and the absolute values of the matrix elements M_{ij} (i <> j) refer respectively to the character and the intensity of the influence of the jth specie upon the ith species, while the M_{ii} is the index of interspecific interaction for the i-th species.

The matrix M, which reflects the structure of the relations in the community, is often termed the community matrix.

The community matrix elements signs follow the rules stated by Odum (E.Odum, Fundamentals of Ecology).

+ stimulating

- **suppressing**
0 **neutral**

The Table 1 reports the pairwise classification, the n-dimensional generalization is straightworward.

Pairwise Interaction		Terminology
+	+	Mutualism or symbiosis
+	-	Predator Prey
+	0	Commensalism
-	-	Competition
-	0	Amensalism
0	0	Indifference (Neutral)

Table 1 Terminology for the 6 pairwise interactions used in the GM mod

c) The modified n species Volterra equations

$$\frac{dx_i}{dt} = x_i\left(\sum_{j=1}^{n} M_{i,j}x_j - r_i + \sum_{l=1}^{p} e_{i,l}u_l(t)\right)$$

Where the x_i, the matrix M and the r_i have the same meaning of the model b) and the e_{il} are the species susceptibility to the time-dependent perturbations $u_l(t)$ (e.g., antibiotic treatment or diet).

3.5 Motivation for stochastic modeling

In biology there are several processes that cannot be described in term of deterministic evolution. From protein production to the behaviour of the whole cell, not only the noise is ever-present, but evolution found several ways to exploit this noise to the advantage of the single cell or the whole population.

One of the most intriguing example of exploitation of the stochasticity is the so-called ``bet hedging strategy'' which can be found in several bacteria population.

The human gut is a very peculiar environment for bacterial growth. It has plenty of nutrients income, but of greatly various kinds; This could severely limit the ability to survive of the bacteria, due to the frequent changes of environmental conditions. Also the competition among different species can be harsh. Bacteria are known to enter a quiescent state to survive to harsher environmental condition. The main problem with this approach is that the process is not instantaneous, but it takes some time to happen (minutes to hours, usually), and this can lead to a nearly 100% extermination of the population in case of rapidly changing environment.

What biologists observed is that in any bacterial population, a fraction of this population is always in the resistant state, whatever the condition was. This allowed a certain percentage of the population to survive no matter how fast the environment fluctuated.

This was true even for monoclonal population and in general a fixed percentage of the population enters this resistant phase regardless of the state of the ancestor. What has been understood is that bacterial cells undergo a transition toward resistant state and back to the reproductive state with a certain fixed probability. The population that doesn't exploit this method will have a competitive advantage in the short term, but in the end will fall victim to the ambient fluctuation, while the oscillating population, albeit slower in growing, will persist to harder perturbations.

The dynamics of a population of individuals can often be represented with a master equation, as the population size is intrinsically a discrete quantity whose evolution in time is driven by random interaction between individuals. Population growth, epidemic diffusion and the evolution, especially in the formulation of neutral theory of evolution, can all be represented on a discrete stochastic basis.

The simplest population growth model treats the individuals as units whose death is a constant process and reproduction is a simple duplication, and it is often used for bacteria with good approximation. A more detailed model that takes into consideration phenomena like male-female interaction, competition for resources and age groups can be written without special difficulties.

The concept of evolution is well known, even if commonly misinterpreted, as a combination of random mutations, both with reproductive fitness advantage or disadvantage, and natural selection, i.e. the inter-specific competition of the individual of the population for resources, mating and avoiding predators. Evolution can also be driven by purely stochastic effects, as shown by Motoo Kimura thirty years ago in his book ``The Neutral Theory of Molecular Evolution'', which launched the concept of neutral evolution. This theory states that in a small population most of the mutations are not fixed in the population by a competitive advantage but rather by mere case, as reproduction can spread a trait among a population and fix by mere fluctuation. The actual probability of fixation for a neutral mutation is in the order of 1/N where N is the population size. Advantageous mutations spread easier and faster while disadvantageous ones spread slower with a higher extinction probability, but still can be fixed if the population is small enough. This idea of neutral evolution is becoming more and more important in biology, as it gives a null hypothesis to test against on evolutionary research.

A similar theory has been developed in the ecological niche distribution among several habitats, and is based on discrete stochastic process starting from simple population dynamic process.

For systems that span more than few molecular species with few hundreds molecules, all the simpler resolution systems fail, one way or the other, due to our limitation in finding analytical solutions in multidimensional system or the limit of the numerical computation, whose rounding errors pile up making any prediction close to meaningless.

In this situation, the only feasible way to analyse a system is through MonteCarlo simulations of the system itself. The main system to perform this simulation is the *Stochastic Simulation Algorithm*, which simulates each reaction step in a painstaking way. The solution obtained with this method has been proved to converge to those of the corresponding master equation. Starting from the original formulation, which is a common workhorse in system biology, several others has been proposed by Gillespie himself to overcome the main limitation of the original algorithm, which is a non bounded time of simulation for stiff systems.

The basic Stochastic Simulation Algorithm is strikingly simple: given a state of the system one has to choose which reaction will happen next between the possible ones and how much time will the system stay still before the reaction happens. This process is iterated until the whole time of interest has been simulated.

Given the slow convergence of this method, two approximation techniques have been developed by Gillespie itself: the tau-leap and the Chemical Langevin Equation.

In the tau-leap one try to evaluate how many reactions take place in a specific time interval tau given the propensity of each reaction at the time t, so that one can approximately use a Poisson distribution to evaluate how many time each reaction fires, and update the system correspondingly.

In the Chemical Langevin Equation, one works with a deterministic ODE for the mean, to which is added a Gaussian noise as in the standard Langevin methods, but designed to respect the correlation of the variation due to the various reactions, extracting one normal variable for each reaction instead of one for each specie as the basic Langevin Equation.

3.6 Application to data analysis

Variability is an intrinsic part of biological samples. Most statistical procedures do not consider this variability as information but rather as an hindrance. This lead to a statistical approach of describing the system with it's expected value, for example with classical ODEs, and the fit to the data is done simply considering the parameter set that minimize the difference between the expected value and the observed ones. More complex approaches, which account for a possible dependency between the expected value and the variability, like the generalized linear model, require a progressively harder mathematical effort in the description of the model.

Stochastic methods have the intrinsic advantage of describing the variability and covariance among observables as a function of the parameters, and no only the expected value. This means that the variability actually carries information on the underlying system, and do no requires any special treatment to be used in the modelling.

Using this conceptual framework the fit to the data is still done as a likelihood maximization procedure on the data. The difference is that the likelihood is not assumed to be a known and simple distribution but rather something that arises naturally from the model and does not require additional tweaking to be considered.

This approach can be also used together with a Bayesian description of the systems. In this statistical framework each parameter is considered a stochastic variable and it's distribution after the observation of the data is given by a combination of the data likelihood and a previous knowledge about the parameter value. This allows us to obtain in a straightforward way a sensitivity analysis for all parameters, with an estimation of cross-sensitivity among different parameters.

d) The stochastic (CME) implementation

The stochastic version of a ODE system can be obtained by the following general

CME:

$$\partial_t P_n(t) = (\mathbb{E}_n^- - 1) g_n P_n(t) + (\mathbb{E}_n^+ - 1) r_n P_n(t)$$

$P_n(t)$ is the probability to have n bacteria at the time t, and E^+ and E^- are the so called Vn Kampen operators defined on the basis of their action on a integer function:

$$\mathbb{E}_n^+ f(n) = f(n+1)$$
$$\mathbb{E}_n^- f(n) = f(n-1)$$

the r_n and g_n terms are the so called recombination and generation terms and are related to the negative and positive terms in the differential equation.

An explicit form for the CME of the equation c) is not very readable, hence we report this in the code for the CME generation (see annex 1).

We remark that an explicit form can be written by separation of the positive terms from the negative terms in the differential equation.

The CME approach (see results) is more precise when we are considering low number of bacteria ($n < 100$). In such cases the effect of fluctuations can be very large and can affect the dynamics. The CME approach is more general of the ODE approach because if we take a larger n, the solution of the CME recovers those of the ODE.

3.7 Technical information

As stated before, all the numerical simulation have been implemented in Python, C and Mathematica. For the sake of code sharing with the other participant, we developed also a prototypical version in Ipython.

The Ipython Notebook is a web-based interactive computational environment where is possible to combine code execution, text, mathematics and graphics into a single document.

We have installed the Ipython on our server and the other project participant can access the module by a remote web client.

The other software will be shared between the participants.

3.7.1 Computing facilities

The Ipython server is hosted in a Linux server with 64 processors and 200 GB of RAM. The integrity of the data and the transmission flow is guaranteed by specific hardware and software infrastructure (firewall, mirroring disks, cryptography etc.). The server has a public section, and a private section, accessible only to members of our group. All the software (R, Mathematica, Matlab, Python, etc.) can also be accessed with a SOAP interface.

4 Deliverable results figures

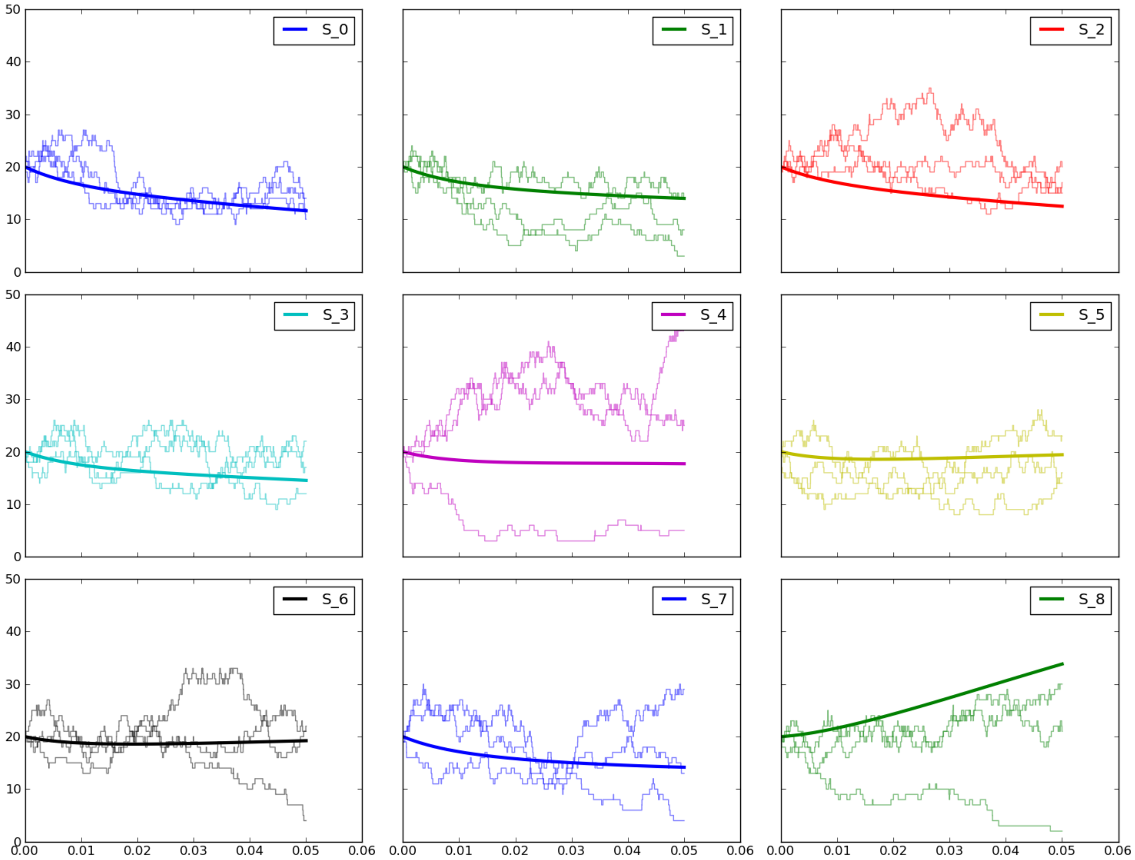

Figure 1 Results for the modified n species Volterra system (n=9). We report the temporal evolution of the deterministic and the stochastic solutions. We report only 3 stochastic solutions, that have been computed in the case of low number of bacteria (it is possible to appreciate the effect of the fluctuations). The time is in arbitrary units. The number of bacteria is approximatively 20. All the species are starting from the same initial condition (e.g.the number of bacteria).

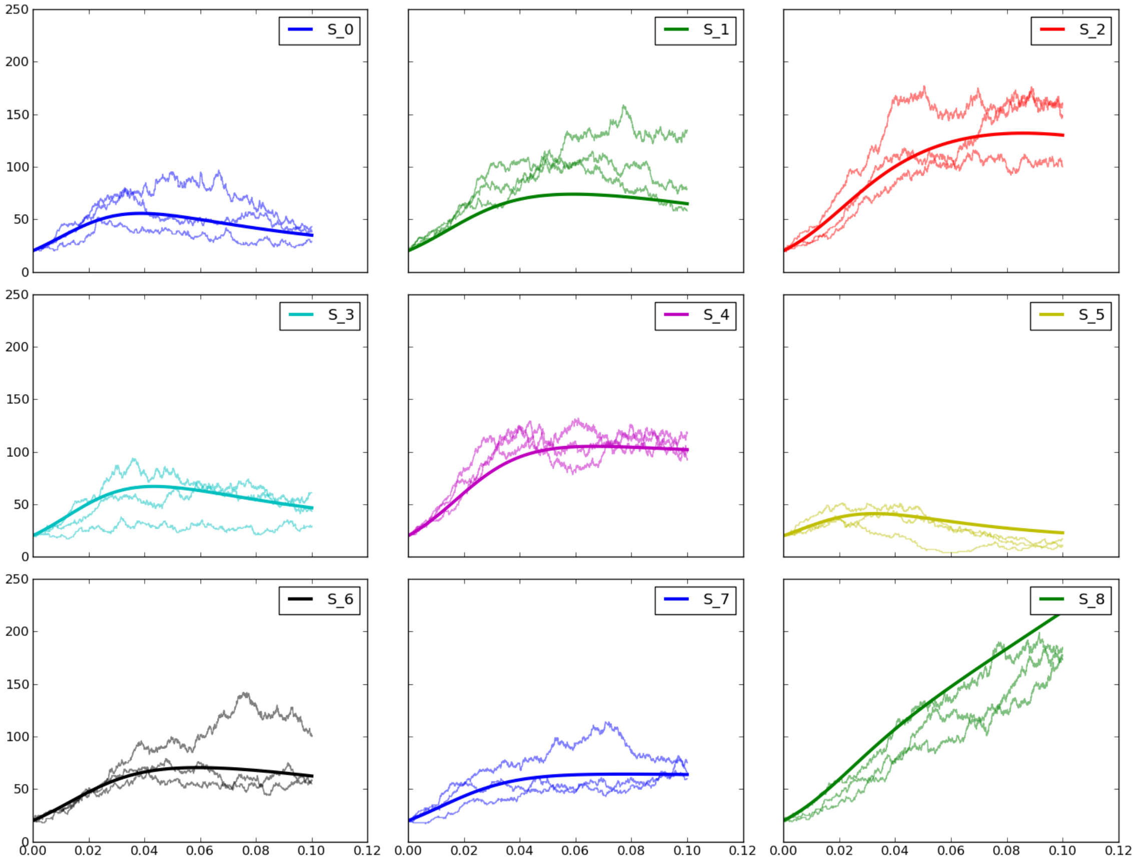

Figure 2 Results for the modified n species Volterra system (n=9). We report the temporal evolution of the deterministic and the stochastic solutions. We report only 3 stochastic solutions, that have been computed in the case of in medium number of bacteria (it is possible to appreciate the reduction of the fluctuations effect). The time is in arbitrary units. The number of bacteria is approximatively 100. All the species are starting from the same initial condition (e.g.the number of bacteria).

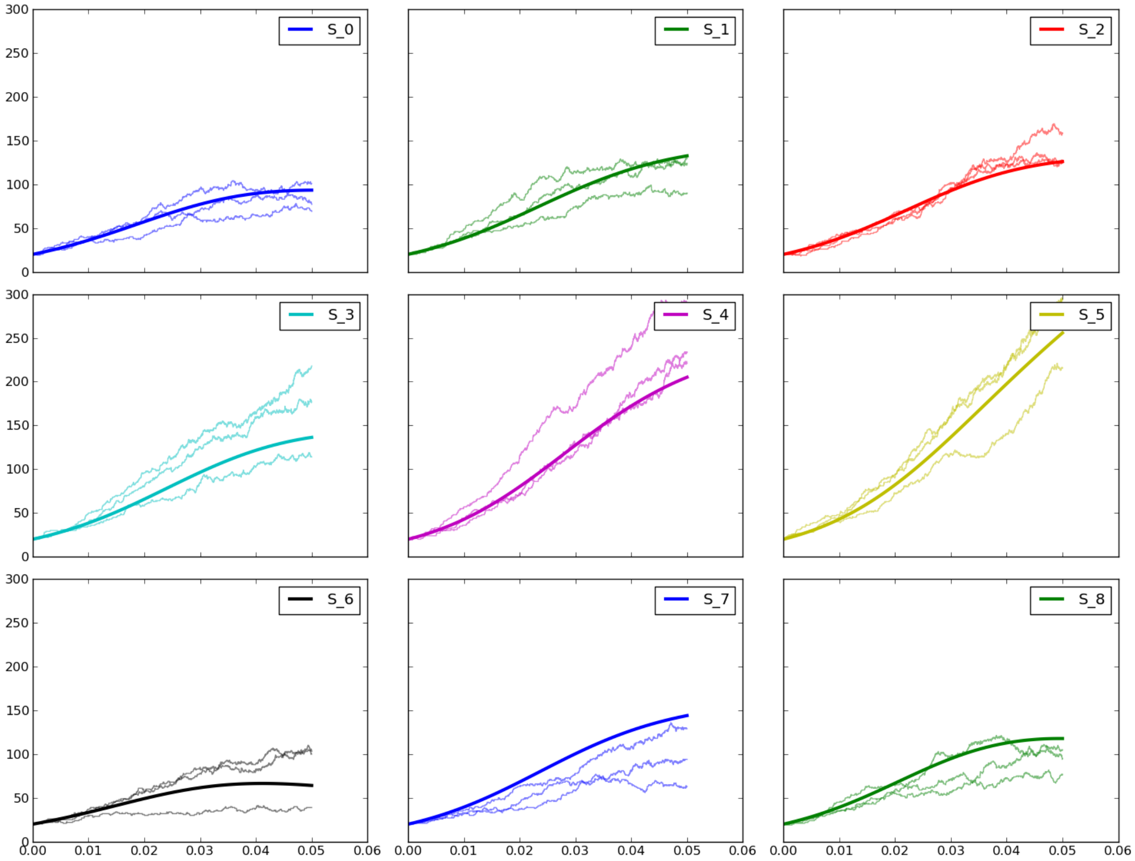

Figure 3 Results for the modified n species Volterra system (n=9). We report the temporal evolution of the deterministic and the stochastic solutions. We report only 3 stochastic solutions, that have been computed in the case of in medium number of bacteria (it is possible to appreciate a further reduction of the fluctuations effect). The time is in arbitrary units. The number of bacteria is approximatively 200. All the species are starting from the same initial condition (e.g.the number of bacteria).

5 Discussion

The dynamical model of Gut Microbiota we created and implemented is important for a series of reasons:

- It takes into account external perturbations such as diet and environmental variations
- This framework can be easily extended to specific genetic background of the host and to intersections with the Gut Microbiota
- This model is and independent module, but it can be a source of input for the Immune System Simulator and can be easily translated in a series of rules, such as

a cellular automata for the Gut Microbiota discrete dynamics.

- The double implementation, deterministic and stochastic is absolutely new and can serve as a powerful basis for data integration, expecially the stochastic one.
- The Chemical Master Equation applied to Gut Microbioma model fournish a rigorous basis for an ecological theory of Metagenomic data, including intersections with Immune System and environment and tor the progression from Insulin resistance to Type 2 Diabetes.
- This model and code will be shared between all the components of this consortium

6 Simulation specification

6.1 Model implementation:

The model was implemented in the Python programming language, leveraging it's capacity for numerical and symbolic calculus. The library used were:

- numpy, for the multidimensional array data structure;
- scipy, for the ODEs integration
- matplotlib, for the plot visualization
- simpy, for the symbolic manipulation needed to evaluate the CME.

The user create an object describing the CMEs to which each reaction is added, specifying the reagent, the products and the kinetics of the reactions.

From this information the program automatically generates the dynamical structures needed both for the Stochastic Simulation Algorithm and the ODEs integration using symbolic algebra manipulation.

All the systems were composed from 9 different species of bacteria and 2 different nutrients were considered.

All the parameters of the models (self growth, interaction and effects of the nutrients) were generated as uniform random in a given interval.

All the bacteria species were initialized to the same value of 20 units, to evidence the different behaviour of different species.

In this model the nutrients were considered a non consumable resource, but inserting periodic insertion would be easy. A little amount of immigration of each species were considered as there is an intake of external bacteria with nutrition, and the model would not be stable without that intake.

Different system size were realized by varying the magnitude of the interaction terms between species, as this is the stronger limit on the population size in these kind of models.

6.2 Deterministic version

The system automatically generates the deterministic ODEs starting from the reaction description of the system.

For example in a mutation even one bacterium transform from the starting genre A to a different one B with a kinetic K. The algorithm use the symbolic algebra to combine all the kinetics for the reagents as negative terms, while it adds as positive terms for the products. It also correctly recognizes cases like reproduction were one bacteria split in two.

The integration is done with the the lsoda algorithm from the

FORTRAN library odepack. This allow us to interrogate the system at the selected point in time, making it really easy to use this simulation in the process of data modeling.

It is important to note that in general ODEs can have a behavior different from the expected values of the master equation, as all the nonlinearities in the model equation will alter the value of the expected values in respect to the deterministic value. This is due to the variance of the model, which the nonlinearities include in the evaluation of the expectation. A simple example of this phenomenon can be seen in the Chi squared distribution, that is the distribution of the square of a normally distributed variables.

This distribution has an expectation equal to the variance of the original distribution, that is very different from the square of the expectation of the normal distribution, which is 0.

These equations also do not consider the possibility of extinction in the evolution of the system.

6.3 Stochastic version

The simulations has been performed using a Stochastic Simulation Algorithm to assure a correct evaluation of the model even for low number of bacteria.

During following phases of the project the model will be implemented as a hybrid algorithm using SSA, tau leaping and Chemical Langevin algorithms to balance precision and time of evaluation. Where the number of molecules is small, the SSA will be used, moving to tau leaping and Chemical Langevin as the numbers grow bigger.

Being a stochastic simulation, what we can obtain is a distribution, ether stationary or evolving in time, to describe our system.

For stationary distribution we can use the ergodic theorem for Markov chain and use a long running simulation to infere the stationary properties.

For the time dependent one the only solution is a massively parallel simulation of several versions of the system to evaluate not only the expected value but also the variance with a sufficient confidence margin

In this phase the model is a single compartment one, but this will be changed in the next phases as we will try to model the internal movements of the gut microbiota.

This is important because each population can suffer from temporary extinction in the stochastic models, whereas this phenomenon cannot happen in the deterministic version. This is always a possibility, but the effect is removed if one consider an immigration, both from the external and other region of the colon.

6.4 Model Testing

Generalized Lotka-Volterra models can exhibits a non-trivial dynamics, as they can in principle describe multistable states. This multistability can emerge when the competition terms are sufficiently strong. As far as we know there is no evidence of this kind of really strong competition between different strand of species, so we worked under the hypothesis that a single stable state exist.

We tested different starting condition for each parameter combinations and observed that the behavior is compatible with a single stable state. This behaviour is clear both on the deterministic ODEs, that are strongly sensible to the initial condition, and the stochastic version, that are robust to differences in the starting position of the distribution.

Given the novelty of the model we started the simulation with a SSA repeated several times on a short timescale, expanding it progressively over a couple of order of magnitude to confirm that what we found was a stable stationary states and not a metastable one.

In general more species were added to the model and faster the convergence to the stationary state, as the increasing number of interaction determined a stronger pressure to reach the equilibrium state.

7 Deliverable Conclusions

We created an ecological model for the GM temporal evolution. The model is based on a adaptation of the classical n- species Volterra equation. We provided both a stochastic and a deterministic version. The deterministic version is very fast, a system of 1000 equation is integrated in few hours, even if the code is not optimized for velocity. An optimized version of code will be developed in the next future. The stochastic version is more slow and its velocity is strongly dependent from the number of bacteria. The behaviour of the system is stable, and each solution reaches a stable state as shown in the results. We surmise that this model will be capable to provide a mechanistic explanation of the species abundance observed in Gut Microbiota data. The model we created is rooted in the conceptualization of Metagenomic (the set of omics measurements aimed to quantify the composition and the interactions dynamics between the host and the microbiome). We collected many metagenomic data, both on GM composition and on host characterization. It will be of great interest to fit this data with our model, both deterministic and stochastic. Starting from some preliminary observation on data fitting we can anticipate that the OTU distribution is a Gamma type in some cases, as predicted by the neutral theory of evolution, but we can observe also variation from this behaviour even if the distributions are with lon tails. This is a very interesting phenomena and will be very important to see how much is conserved across our data.

8 Annex: Simulation Code

```
# -*- coding: utf-8 -*-
# <nbformat>3.0</nbformat>
# <codecell>

from __future__ import division
#from sympy import *
import sympy
from sympy import Symbol
from scipy.integrate import odeint


# <codecell>

from collections import defaultdict
#from IPython.core.display import display

# <codecell>

import numpy as np
from numpy.random import exponential as rand_exp
#import pylab

from sympy.utilities.lambdify import implemented_function, lambdify

# <codecell>

class myCounter(defaultdict):
    def __init__(self, other={}):
        defaultdict.__init__(self, int)
        for k, v in other.items():
            self[k]+=v
    def __add__(self, other):
        new_counter = myCounter(self)
        for k, v in other.items():
            new_counter[k]+=v
        return new_counter

    __radd__ = __add__

    def __sub__(self, other):
        new_counter = myCounter(self)
        for k, v in other.items():
            new_counter[k]-=v
        return new_counter

    def __rsub__(self, other):
        new_counter = myCounter(other)
        for k, v in self.items():
            new_counter[k]-=v
        return new_counter

    def __mul__(self, other):
        new_counter = myCounter()
        for k, v in self.items():
            new_counter[k]+= other * v
        return new_counter

    __rmul__ = __mul__
    def positive(self):
        return all(v>=0 for v in self.values())

    def __str__(self):
        return "{"+ ", ".join("{}={}".format(k,v) for k, v in self.items()) +"}"

    __repr__ = __str__

# <codecell>

def variazione(expr):
    """given an expression returns the corresponding variation of the state
    A   --->  {A:1}
    A+B --->  {A:1, B:1}
    2*A --->  {A:2}
    """
    res = myCounter()
    if expr is None:
        return res
    for s in expr.free_symbols:
        res[s] = sympy.diff(expr, s)
    return res

# <codecell>

def shift(state, substrate, products, kinetic):
    """given a starting state and a variation on the state,
    it returns the destination state and the transition constant or None
    """
    first_passage = state - substrate
    if first_passage.positive():
        return first_passage + products, kinetic.subs(state)
    else:
        return None, None

# <codecell>

class CME(object):
    def __init__(self):
        self.reactions = []

    def add_reaction(self, substrate, products, kinetic):
        """add a reaction to the CME, given the consumed substrate, the created product and the
reaction kinetic"""
        self.reactions.append( (variazione(sympy.sympify(substrate)),
                                variazione(sympy.sympify(products)),
                                sympy.sympify(kinetic)) )

    def escapes(self, start):
        """given a starting state it evaluate which states are reachable and the corresponding
transition rate"""
        start = myCounter(start)
        end_states = []
        kinetics = []
        for substrate, products, kinetic in self.reactions:
            end_state, kinetic = shift(start, substrate, products, kinetic)
            if kinetic and end_state is not None:
                end_states.append(end_state)
                kinetics.append(float(kinetic))
        kinetics = np.array(kinetics)
        return end_states, kinetics

    def gillespie(self, start, t_end=10.0):
        """make n step of gillespie simulation given the starting state"""
        start = myCounter(start)
        steps = t_end
        time = 0.0
        while time<steps:
            end_states, kinetics = self.escapes(start)
            cumulative = np.cumsum(kinetics)
            if not len(end_states) or not len(cumulative):
                # ho raggiunto uno stato stazionario
                yield start, np.inf
                break
            lambda_tot = cumulative[-1]
            dt = rand_exp(1./lambda_tot)
            selected = np.searchsorted(cumulative/lambda_tot, np.random.rand())
            new_state = end_states[selected]
            yield start, dt
            time +=dt
            start = new_state

    def evaluate(self, start, t_end, *functions):
        """evaluate the value of several function in time given a starting state and the number of
step to be done"""
        time = 0.0
        states, dts = zip(*self.gillespie(start, t_end))
        time = np.cumsum([0] + list(dts))
        states = [start] + list(states)
        func_values = { function:[ float(function.subs(state)) for state in states ] for function in
functions}
        return time, func_values

    def distribution(self, start, steps=10, burnout=-1, *functions):
        """return the stationary distribution from a gillespie simulation"""
        distrib = defaultdict(float)
        time = 0.0
        for idx, (state, dt) in enumerate(self.gillespie(start, steps)):
            time+=dt
            if time>burnout:
                state = tuple(sorted(state.items(), key=str))
                distrib[state]+=dt
        if not distrib:
            return {tuple(sorted(state.items(), key=str)):1.0}
        result = {}
        for function in functions:
            if isinstance(function, (tuple,list)):
                pass
            else:
                A_distrib = { int(function.subs(dict(k))):v for k, v in distrib.items()}
                min_a, max_a = min(A_distrib), max(A_distrib)
                A_distrib = [A_distrib.get(idx, 0.0) for idx in xrange(min_a, max_a+1)]
                result[function] = A_distrib
        return result
    def writeCME(self):
        """write the complete CME of the given process"""
        p = Symbol('p')
        pxy = p(*sorted(k for k in set.union(*[set(substrate-products) for substrate, products, kinetic
in self.reactions])))
        base = 0
        for substrate, products, kinetic in self.reactions:
            transition = substrate-products
            temp = (pxy*kinetic).subs( {k: k+transition.get(k, 0) for k in transition}) - pxy * kinetic
            base += temp
        return base

    def transition_matrix(self, start):
        """create the transition matrix and the state vector
        from a starting point

        Will stuck in an infinite loop if the CME is not limited
        """
        start = myCounter(start)
        states = [start]
        transitions = dict()
        for state in states:
            for destination, kinetic in zip(*self.escapes(state)):
                if destination not in states:
                    states.append(destination)
                transitions[tuple(state.items()),
                            tuple(destination.items())] = kinetic
        return transitions, states

    def MCMCstep(self, start):
        for result, time in self.gillespie(start, steps=1.0):
            pass
        return result

    def symbol_set(self):
        symbol_set = set()
        for reaction in self.reactions:
            prod, reag, kine = reaction
            for k in prod.keys():
                symbol_set.add(k)
            for k in reag.keys():
                symbol_set.add(k)
            for k in [s for s in kine.atoms() if s.is_Symbol]:
                symbol_set.add(k)
        return sorted(symbol_set)

    def odeint(self, start, time=(0.0, 1.0)):
        symbols = self.symbol_set()
        x0 = [start[s] for s in symbols]
        modifiche = { s:0.0 for s in symbols }
        for reaction in self.reactions:
            prod, reag, kine = reaction
            for k, v in prod.items():
                modifiche[k] -= v*kine
            for k, v in reag.items():
                modifiche[k] += v*kine
        funzioni = {k:lambdify(symbols, f) for k, f in modifiche.items()}
        funzioni = [funzioni[s] for s in symbols]
        def derivata(values, t):
            return [f(*values) for f in funzioni]
        res = odeint(derivata, x0, time).T
        return {s:res[i] for i,s in enumerate(symbols)}


if __name__ == '__main__':
    import pylab
    from itertools import combinations_with_replacement as cwr
    from sympy import Rational as R

    species = sympy.symarray('S', 9)
    nutrients = sympy.symarray('N', 2)

    symbols =    list(species)+list(nutrients)

    print "inizio a creare la cme"

    cme = CME()
    print "\tinserisco termini singoli"
    for specie in species:
        # decadimento del batterio
        k = R(10+pylab.randint(20), 200)
        cme.add_reaction(specie,
                         None,
                         k*specie)

    print "\tinserisco termini di competizione"
    for specie_1, specie_2 in cwr(species, 2):
        k = R(10+pylab.randint(20), 200)
        cme.add_reaction(specie_1,
                         None,
                         k*specie_1*specie_2)
        k = R(10+pylab.randint(20), 200)
        cme.add_reaction(specie_2,
                         None,
                         k*specie_1*specie_2)

    print "\tinserisco termini dei nutrienti"
    for specie in species:
        for nutrient in nutrients:
            k = R(20+pylab.randint(20), 10)
            # immigrazione del batterio 1 per via del cibo
            cme.add_reaction(None,
                             specie,
                             k*nutrient)
            # riproduzione del batterio 1 per via del cibo
            k = R(30+pylab.randint(20), 10)
            cme.add_reaction(None,
                             specie,
                             k*nutrient*specie)
            # morte del batterio 1 per via del cibo
            #cme.add_reaction(specie,
            #                   None,
            #                   (0.1+pylab.rand())*nutrient*specie)
```

```
start_state = {}
for specie in species:
    start_state[specie] = 20
for nutrient in nutrients:
    start_state[nutrient] = 10

L = len(species)
r, c = 3, 3
fig, ax = pylab.subplots(r, r,
                         figsize=(5*r, 4*c),
                         sharex=True,
                         sharey=True)
assi = {s:axi for s, axi in zip(species, ax.ravel())}

colors = {}
color_cycle = assi.values()[0]._get_lines.color_cycle
for s, c in zip(symbols, color_cycle):
   colors[s]=c

print "inizio i gillespie"
times = []
time_end = 0.1
simulazioni = 3
for i in xrange(simulazioni):
    print "\tinizio simulazione numero {} di {}".format(i+1, simulazioni)
    time, functions_dict = cme.evaluate(start_state, time_end, *symbols)
    last_time = time[-1] if not isinf(time[-1]) else time[-2]*1.1
    times.append(last_time)
    for symbol in species:
        assi[symbol].plot(time,
                functions_dict[symbol],
                linestyle='steps-mid',
                alpha=0.5,
                color=colors[symbol])

#ax.set_xlim(0.0, 1.0)

tempo = pylab.r_[0.0: max(times): 100j]

print "inizio la deterministica"

res = cme.odeint(start_state, tempo)
for specie in species:
    k = specie
    v = res[k]
    assi[k].plot(tempo, v,
            label=str(k),
            color=colors[k],
            linewidth=3)

print "terminato"

for asse in ax.ravel():
    asse.legend()
fig.tight_layout()
```

9 Bibliography

Stein RR, Bucci V, Toussaint NC, et al. Ecological modeling from time-series inference: insight into dynamics and stability of intestinal microbiota. PLoS Comput Biol. 2013;9(12):

Karlsson FH, Nookaew I, Petranovic D, Nielsen J. Prospects for systems biology and modeling of the gut microbiome. Trends Biotechnol. 2011;29(6):251–8.

Yu M. Svirezhev; Dmitrii O. Logofet Stability of Biological Communities Published by MIR Publishers, 1983 ISBN 10: 0828523711 / ISBN 13: 9780828523714

Lozupone CA, Stombaugh JI, Gordon JI, Jansson JK, Knight R (2012) Diversity, stability and resilience of the human gut microbiota. Nature 489: 220–230.

Relman DA (2012) The human microbiome: ecosystem resilience and health. Nutr Rev 70: S2–S9.

Dethlefsen L, Relman DA (2011) Incomplete recovery and individualized responses of the human distal gut microbiota to repeated antibiotic perturbation. Proc Natl Acad Sci 108: 4554–4561.

Jernberg C, Löfmark S, Edlund C, Jansson J (2007) Long-term ecological impacts of antibiotic administration on the human intestinal microbiota. ISME J 1: 56–66.

Allison SD, Martiny JB (2008) Resistance, resilience, and redundancy in microbial communities. Proc Natl Acad Sci 105: 11512–11519.

Faust K, Sathirapongsasuti JF, Izard J, Segata N, Gevers D, et al. (2012) Microbial co-occurrence relationships in the human microbiome. PLoS Comput Biol 8:

Bucci V, Bradde S, Biroli G, Xavier JB (2012) Social interaction, noise and antibiotic-mediated switches in the intestinal microbiota. PLoS Comput Biol 8: e1002497. doi: 10.1371/journal.pcbi.1002497

Gerber GK, Onderdonk AB, Bry L (2012) Inferring dynamic signatures of microbes in complex host ecosystems. PLoS Comput Biol 8: e1002624. doi: 10.1371/journal.pcbi.1002624

Mounier J, Monnet C, Vallaeys T, Arditi R, Sarthou AS, et al. (2008) Microbial interactions within a cheese microbial community. Appl Environ Microbiol 74: 172–181. doi: 10.1128/aem.01338-07

Hofbauer J, Sigmund K (1998) Evolutionary games and population dynamics. Cambridge University Press.

May RM (2001) Stability and complexity in model ecosystems, volume 6. Princeton University Press.

Yeung MKS, Tegnér J, Collins JJ (2002) Reverse engineering gene networks using singular value decomposition and robust regression. Proc Natl Acad Sci 99: 6163–6168. doi: 10.1073/pnas.092576199

Gardner TS, Di Bernardo D, Lorenz D, Collins JJ (2003) Inferring genetic networks and identifying compound mode of action via expression profiling. Science 301: 102–105. doi: 10.1126/science.1081900

Bonneau R, Reiss DJ, Shannon P, Facciotti M, Hood L, et al. (2006) The inferelator: an algorithm for learning parsimonious regulatory networks from systems-biology data sets de novo. Genome Biol 7: R36. doi: 10.1186/gb-2006-7-5-r36

Bansal M, Della Gatta G, Di Bernardo D (2006) Inference of gene regulatory networks and compound mode of action from time course gene expression profiles. Bioinformatics 22: 815–822. doi: 10.1093/bioinformatics/btl003

White JR (2010) Novel Methods for Metagenomic Analysis. Ph.D. thesis, University of Maryland.

Faust K, Raes J (2012) Microbial interactions: from networks to models. Nature Rev Microbiol 10: 538–550. doi: 10.1038/nrmicro2832

Tikhonov A, Arsenin VY (1977) Solution of Ill-posed Problems. VH Winston & Sons.

Aster RC, Borchers B, Thurber CH (2012) Parameter estimation and inverse problems. Academic Press.

Zeeman ML (1995) Extinction in competitive Lotka–Volterra systems. Proc Amer Math Soc 123: 87–96. doi: 10.1090/s0002-9939-1995-1264833-2

Kim JG (1996) Coexistence in competitive Lotka–Volterra systems. Comm Kor Math Soc 11: 147–151.

Lupton JR, Ferrell RG (1986) Using density rather than mass to express the concentration of gastrointestinal tract constituents. J Nutr 116: 164–168.

Vano J, Wildenberg J, Anderson M, Noel J, Sprott J (2006) Chaos in low-dimensional Lotka–Volterra models of competition. Nonlinearity 19: 2391. doi: 10.1088/0951-7715/19/10/006

Amann H (1990) Ordinary differential equations: an introduction to nonlinear analysis, volume 13. de Gruyter.

Shea K, Chesson P (2002) Community ecology theory as a framework for biological invasions. Trends Ecol Evol 17: 170–176. doi: 10.1016/s0169-5347(02)02495-3

Shade A, Peter H, Allison SD, Baho DL, Berga M, et al. (2012) Fundamentals of microbial community resistance and resilience. FMICB 3: 417. doi: 10.3389/fmicb.2012.00417

Dai L, Vorselen D, Korolev KS, Gore J (2012) Generic indicators for loss of resilience before a tipping point leading to population collapse. Science 336: 1175–1177. doi: 10.1126/science.1219805

Connell JH, Sousa WP (1983) On the evidence needed to judge ecological stability or persistence. Amer Nat 121: 789–824. doi: 10.1086/284105

Werner JJ, Knights D, Garcia ML, Scalfone NB, Smith S, et al. (2011) Bacterial community structures are unique and resilient in full-scale bioenergy systems. Proc Natl Acad Sci 108: 4158–4163. doi: 10.1073/pnas.1015676108

Turnbaugh PJ, Ley RE, Hamady M, Fraser-Liggett CM, Knight R, et al. (2007) The human microbiome project. Nature 449: 804–810. doi: 10.1038/nature06244

Stecher B, Chaffron S, Käppeli R, Hapfelmeier S, Freedrich S, et al. (2010) Like will to like: abundances of closely related species can predict susceptibility to intestinal colonization by pathogenic and commensal bacteria. PLoS Pathog 6: e1000711. doi: 10.1371/journal.ppat.1000711

Barenblatt GI (1996) Scaling, self-similarity, and intermediate asymptotics: dimensional analysis and intermediate asymptotics, volume 14. Cambridge University Press.

Bishop CM (2006) Pattern recognition and machine learning. Springer New York.

Fernando Pérez, Brian E. Granger, *IPython: A System for Interactive Scientific Computing*, Computing in Science and Engineering, vol. 9, no. 3, pp. 21-29, May/June 2007, doi:10.1109/MCSE.2007.53. URL: http://ipython.org

Daniel Remondini, Enrico Giampieri, Armando Bazzani, Gastone Castellani & Amos Maritan Analysis of noise-induced bimodality in a Michaelis–Menten single-step enzymatic cycle, Physica A 392(2), 336–342 (2012)

Animesh Agarwal, Rhys Adams, Gastone C. Castellani, and Harel Z. Shouval “On the precision of quasi steady state assumptions in stochastic dynamics”, J Chem Phys. 2012 Jul 28;137(4):044105.

A.Bazzani, G. Castellani, E.Giampieri, D. Remondini, LN Cooper “Bistability in the Chemical Master Equation for Dual Phosphorylation Cycles” J Chem Phys. 2012 Jun 21;136(23):235102.

Giampieri E, Remondini D, de Oliveira L, Castellani G, Lió P. Stochastic analysis of a miRNA-protein toggle switch. Mol Biosyst. 2011 Oct1;7(10):2796-803

J. Elf, K. Nilsson, T. Tenson, , and M. Ehrenberg, Bistable bacterial growth rate in response to antibiotics with low membrane permeability, Phys. Rev. Lett., 97 (2006), pp. 258104 1–4.

M. Kimura, The Neutral Theory of Molecular Evolution, Cambridge University Press, 1983.

I. Volkov1, J. R. Banavar1, S. P. Hubbell, and A. Maritan, Patterns of relative species abundance in rainforests and coral reefs, Nature, 450 (2007), pp. 45–49.

F. He Deriving a neutral model of species abundance from fundamental mechanisms of population dynamics,Functional Ecology,p. 187ˆ a193.19 (2005),

Castellani GC, Bazzani A, Cooper LN. “Toward a microscopic model of bidirectional synaptic plasticity.” Proc Natl Acad Sci U S A. 2009 Aug 18;106(33):14091-5.