# ENISI SDE: A Novel Web-based Stochastic Modeling Tool for Computational Biology

Yongguo Mei, Adria Carbo, Raquel Hontecillas, and Josep Bassaganya-Riera
The Center for Modeling Immunity to Enteric Pathogens
Nutritional Immunology and Molecular Medicine Laboratory
Virginia Bioinformatics Institute, Virginia Tech, Blacksburg, VA 24061, USA
{ymei, acarbo, rmagarzo, jbassaga}@vbi.vt.edu

***Abstract*—Stochasticity is part of the nature of many biological processes and an important aspect in modeling and simulations of computational biology. Gillespie's algorithm was developed for modeling stochasticity of chemical reactions and has been broadly applied to stochastic modeling in computational biology. The Gillespie's algorithm accounts for the particle effect in chemical reactions. However, many biological processes including cellular and molecular immunological mechanisms have many other sources of stochasticity such as cell movement, ligand binding, or unaccounted variation in experimental settings. In this paper, we propose stochastic differential equations (SDE) being used as a generic stochastic modeling technique for systems immunology. SDE has been widely used in statistics and economics areas; but only a few isolated studies in computational biology are found using SDE to model stochastic behaviors of cells and molecules. In addition, to the best of our knowledge, there is no user-friendly SDE-based modeling tool available for computational biologists. This paper presents ENISI SDE, a web-based user-friendly stochastic modeling tool for computational biologists. This work provides three major contributions: (1) we discuss SDE and propose it as a generic approach for stochastic modeling in computational biology; (2) we develop ENISI SDE, a web-based user-friendly SDE modeling tool that only requires little extra effort beyond regular ODE-based modeling; (3) we use the model SDE modeling tool to study stochastic sources of cell heterogeneity in the context of a CD4+ T Cell differentiation process. The case study clearly shows the effectiveness of SDE as a stochastic modeling approach in biology in general and immunology in particular and the power of the SDE modeling tool we developed.**

## I. Introduction

Results from biological experiments often vary significantly due to certain degree of randomness associated with them. For example, the same *in vivo* experiment conducted under the exactly same conditions in distinct animal can have significantly different results. On the one hand, experimentalists can better control the experimental settings to reduce the variation; on the other hand, stochasticity is intrinsic to many biological processes including cell movement, cell heterogeneity and ligand binding and thus impossible to eliminate. Indeed in some cases, stochasticity is benign and essential. For example, Schneidman et. al. [31] showed that the stochasticity of neuron ion channel may be critical in determining the reliability and precision of spike timing.

Modeling and simulation techniques have been widely adopted in computational biology. Biological systems are highly complex, nonhomegeneous, have many feedback loops and modeling and simulations can be of great value in helping to understand biological processes:

- Network models can synthesize and represent complex existing knowledge of biological systems.
- Effective reasoning and knowledge discovery can be achieved through applying advanced network inference algorithms onto the models.
- Performing *in silico* experiments, i.e., computational simulations, based upon models can test and refine novel hypotheses for further wet-lab experiments and save significant time and cost.

The most commonly used modeling techniques are equation-based and agent-based models. Equation-based models represent a complex system with a set of mathematical equations. Common equation-based models include ODE (ordinary differential equation) and PDE (partial differential equation) models. There are many efficient numerical computation techniques for equation-based models. The agent-based models use object-oriented programming techniques to represent simulated entities as individual agents or objects. It can simulate more fine-tuned individual behaviors and interactions between agents when compared with equation-based models. Agent-based models use simple rules to model agent individual behaviors and their interactions and can be used to capture highly complex system behaviors. Compared with equation-based techniques, agent based modeling usually requires more computational resources.

Deterministic modeling techniques capture the average group behavior very well; with the same settings a deterministic model always gives you the same results. To account for various stochasticities in biological processes, stochastic modeling is increasingly becoming a necessity. While agent-based models can incorporate stochasticity easily by adding it into the rules, adding stochasticity into equation-based models is generally more challenging. An explanation for that is that by adding stochasticity into the equations, many effective numeric algorithms no longer work as efficiently as in the deterministic cases. In the past decades, there have been many deterministic equation-based models developed in life sciences; however, there are relatively few stochastic models.

Gillespie's algorithm [12] and its variants [5] [29] [4] have been developed for modeling stochasticity in chemical reactions that are represented by ODEs. Gillespie's algorithm captures the stochasticity primarily from the particle effects of molecules or atoms in chemical reactions. The Gillespie's algorithm has been extensively used in simulating many biochemical reactions such as in metabolic pathways. However, Gillespie's algorithm has its disadvantages. First, it is computationally complex and thus expensive. In fact, many models have to reduce the scale of concentrations or the number of particles to get the Gillespie's algorithm to work. Second, it does not capture stochasticity due to other effects.

For ODE-based modeling, there are many tools including Matlab, R, COPASI [15], and others. COPASI was developed originally for biochemical networks and later it has been extended as a general ODE-based modeling platform. COPASI has successfully been applied to signal transduction networks, cellular metabolic networks, and gene regulatory networks. The targeted users of COPASI were originally chemists and biochemists, although they have evolved in the last year to include computational immunologists. As a result, COPASI has highly usable quality interfaces and users need no advanced mathematical knowledge to develop models and perform simulations. In contrast, Matlab is a platform for engineers and R for statisticians and they are not as user-friendly as COPASI for biologists. To the best of our knowledge, for stochastic modeling of ODE models, most of tools include only Gillespie's algorithm and its variants.

Stochastic differential equations (SDE) for stochastic modeling has been widely used in economics and statistics [17]. However, only a few studies [22] [23] [7] [30] [24] applied SDE into computational biology modeling. For those studies, the models are relatively small in scale with only a few ODEs and some of them used Matlab to directly build the SDEs. We have found one SDE package in Matlab [32] and one in R [17]; however, they are not user-friendly, specially when biologists with limited computational training want to use them.

In this study, we have developed ENISI SDE, a web-based user-friendly SDE modeling tool. In the front end, a web browser takes an ODE model file and the stochasticity settings and sends them to the backend server. The backend server runs COPASI, an open source platform developed at Virginia Bioinformatics Institute, and R, a widely accepted computational platform, to perform the SDE simulations. The results are sent back to the web browser through AJAX APIs. To our best knowledge, this tool is the first SDE modeling tool designed for biological applications with the underlying mathematics transparent to users. We have applied this tool to a CD4+ T Cell differentiation model with 93 species and 46 reactions [6]. This tool requires a minimal effort beyond what is required for ODE-based modeling. The biologists can quickly understand and play with the SDE models using this new tool. User-friendly SDEs represent powerful generic stochastic modeling techniques for computational biology. Users are encouraged to use our web-based tool to develop SDE models and perform simulations.

## II. Related Work

Mathematical modeling and simulations have a long history in computational biology [13] [9] [25] [8] [34]. Especially since the introduction of systems biology [19] [16], systems-level modeling techniques [3] [18] are becoming more important in analyzing data, discovering new knowledge, and performing *in silico* experiments in a time and cost saving manner.

ODE-based and agent-based [10] are two of the most popular types of computational modeling used by biologists. Novak et. al. used ODEs to model cell division [26] and DNA replication [27] in fission yeast. Holcombe et. al. presented an agent-based modeling tool called FLAME [14] in modeling complex biological systems. Adra et. al. developed a multi-scale modeling of human epidermis system [1].

For stochastic modeling of ODE-based models, the Gillespie's algorithm [11] [12] has been widely used. For example, Jong et. al. [8] discussed various modeling techniques used for gene regulatory networks and focused primarily on Gillespie's algorithm in the section of stochastic modeling. However, Gillespie's algorithm is computationally complex even though there have been several variants [5] [29] [4] [20] [21]. The Gillespie's algorithm of stochastic modeling has several advantages:

- It has solid theoretic foundation and has been used successfully for modeling chemical reactions.
- Many ODE-based modeling tools provide Gillespie's algorithm.
- It requires little extra effort in addition to ODE-based modeling.

However, the Gillespie's algorithm primarily captures the stochasticity stemming from particle effects of chemical reactions. In contrast, SDE as a stochastic modeling technique has been widely studied and applied in statistics and business [17]. There have been a few isolated SDE models that have been developed for computational biology [22] [23] [7] [30] [24] . To the best of our knowledge, there is no user-friendly tool available that provides SDE modeling capability. For example, Saarinen [30] developed their own private Matlab-based SDE modeling module. Developing large-scale ODE-based models is difficult in Matlab and it is even more challenging for SDE modeling since the users need to write their own mathematical equations in Matlab. This is not feasible for biologists with limited mathematical and computational training.

## III. Stochastic Modeling with SDE

### A. SDE and the Stochastic Process

A stochastic differential equation can be divided into two parts: regular ODE and noise as shown in equation 1, where $X_t$ as a variable of time $t$ is a vector of the concentrations of species. Not considering noise, $G(X_t)$ is the changing rate or derivative of $X_t$ in terms of time.

$$dX_t = G(X_t) * dt + dW_t \tag{1}$$

The noise part, $W_t$ is a vector of stochastic processes. In the rest of this section, we focus on a one dimensional version of equation 1, i. e., $dx_t = g(X_t) * dt + dw_t$ for a detailed theoretical analysis. We assume changing rate of $x_t$ depends on the vector $X_t$ through reactions and its own stochastic noise $w_t$.

The $w_t$ is a stochastic process with the following properties:

1) The mean of $w_t - w_s$ is 0 where $t$ and $s$ are two arbitrary time points.
2) The variance of $w_t - w_s$ is $(t - s)\sigma^2$, where $\sigma$ is a constant.
3) The two random variables $(w_{t2} - w_{t1})$ and $(w_{t4} - w_{t3})$ are independent for any four time points that satisfy $t1 < t2 \leq t3 < t4$.

### B. Numeric Algorithm

*Theorem 1:* If $dw_t$ equals $n_t * \sqrt{dt}$, where $n_t$ is a random variable of mean $0$ and variance $\sigma^2$, i. e., standard deviation of $\sigma$, and $n_t$ is independent of $n_s$ if $t \neq s$, then $w_t$ satisfy the above three properties.

*Proof:* According to the definition, $w_t - w_s = \int\limits_s^t dw$, where $dw = \lim_{\delta t \to 0}(w_{t+\delta t} - w_t)$. The $mean(dw)$ is $mean(n_t) * \sqrt{dt} = 0$ and $variance(dw)$ is $variance(n_t) * dt = \sigma^2 * dt$. For independent random variables, the mean of the sum of random variables is the sum of the means of individual random variable. Therefore, $mean(w_t - w_s) = \int\limits_s^t mean(dw) = 0$. This proves the property 1. The variance of the sum of independent random variables is the sum of the variances of individual random variable. Therefore, $variance(w_t - w_s) = \int\limits_s^t variance(dw) = \int\limits_s^t \sigma^2 dt = (t - s)\sigma^2$. This proves the property 2. Since all $dw_t$ are independent, $w_{t2} - w_{t1}$ and $w_{t4} - w_{t3}$ are independent for any four time points that satisfy $t1 < t2 \leq t3 < t4$ since the sums of two exclusive groups of independent variables are independent. This proves property 3. ■

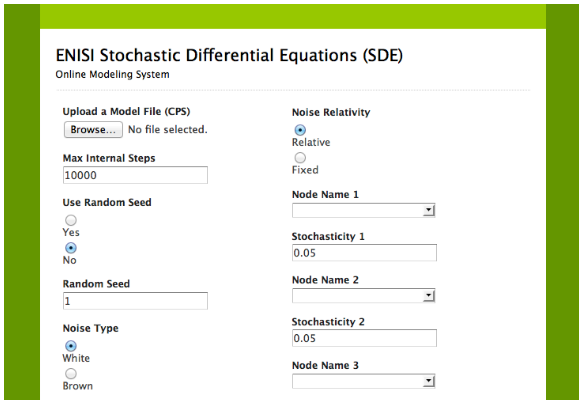
Theorem 1 provides the theoretical foundation for our numerical algorithm. Our numerical algorithm for SDE is divided into two parts. In each step of time progress $dt$, part one is the LSODA (Livermore Solver of Ordinary Differential Equations) algorithm [28] for the ODE integration part $g(X_t)dt$, and part two is adding the stochastic variations $dw_t = n_t\sqrt{dt}$. The random variable $n_t$ is of mean 0 and variance $\sigma^2$.

### C. Discussions on the Stochasticity

Two items remain for precisely defining the noise random variable $n_t$: the noise/distribution type and the standard deviation $\sigma$. The stochastic noise $n_t$ could be white, brownian, or others. The white noise follows uniform distribution where the probability density function (pdf) is const within the distribution range. The one dimensional pdf is shown in equation 2 where a and b are low and high bounds. Setting $b = -a = \sqrt{3}\sigma$, the mean of $n_t$ is 0, and the variance is $\sigma^2$.

$$f(n) = \begin{cases} 1/(b-a) & \text{if a} \leq \text{n} \leq \text{b} \\ 0 & \text{if n} < \text{a or n} > \text{b} \end{cases} \tag{2}$$

The Brownian noise follows the Gaussian distribution and the one-dimensional PDF is shown in equation 3 where $\mu$ is the mean and $\sigma$ is the standard deviation. We just need to set the $\mu$ to 0 for $n_t$.



ENISI Stochastic Differential Equations (SDE)

Online Modeling System

Upload a Model File (CPS)

No file selected.

Max Internal Steps: 10000

Use Random Seed: ○ Yes ● No

Random Seed: 1

Noise Type: ● White ○ Brown

Noise Relativity: ● Relative ○ Fixed

Node Name 1:

Stochasticity 1: 0.05

Node Name 2:

Stochasticity 2: 0.05

Node Name 3:

Fig. 1. The web user interface of the SDE modeling tool.

$$f(n) = \frac{1}{\sigma\sqrt{2\pi}} e^{\frac{1}{2}(\frac{n-\mu}{\sigma})^2} \tag{3}$$

To specify the standard deviation $\sigma$, we assume it is proportional to the corresponding species' concentration $x$, i.e., $\sigma = \theta x$ where $\theta$ is a constant. In each numerical step, the $\sigma$ can be chosen to be proportional to the species' initial concentration $x_0$ or the dynamic concentration $x_t$. We call the former "fixed" and the latter "relative" as the $\sigma$ is fixed during the simulation if choosing to be proportional to the initial concentration. Generally, $\theta$ is a number between 0 and 1. In rare cases of the Brownian noises, the noise could be a very large negative number and the concentration could be negative after adding the noise. For practical reasons, we do not allow negative concentrations of species and set the concentration to be 0 when that happens.

According to the central limit theory, when the simulation interval is small and the number of steps is large, it does not make much difference whether the noise is white or brownian, as the sum of the large number of uniform random variables also approximates a Gaussian distribution.

## IV. ENISI SDE, a Web-Based Modeling Tool

### A. Web-Based Front End

Figure 1 shows a partial snapshot of the web user interface we developed for SDE modeling. This is the front end of the tool and is used for capturing user inputs to the model and presenting the results. Users only need a web browser to use the SDE modeling tool. Making the interface web-based largely eliminates hardware and operating system dependencies that force many tools to provide multiple versions for various systems such as Linux or Windows. Our tool is also tablet and smartphone friendly since it requires only a web browser.

In terms of the implementation, the front end is implemented with HTML and JavaScripts. HTML is for the static part and JavaScripts for the dynamic part. The jQuery library and AJAX apis are used in JavaScripts. AJAX technology allows the user interface to wait for the backend computations and present the results to the specific section of the web page without reloading the whole page.

The users need to provide a valid ODE model file in XML-based COPASI format. For ODE model development, please refers to the COPASI user manual for more details. In addition t the ODE model file, users also need to provide the following SDE specific parameters:

- Providing a random seed. When users provide the random seed, the results can be reproducible. If leaving the UseRandomSeed as default 0/disabled, the system will automatically generate random seed based upon system time.
- Select noise type whether it is white or brownian as explained earlier. The default is white.
- Select whether the noise is relative or fixed explained earlier. The default is relative.
- Specify the node name and the corresponding proportion or $\theta$. For example, specifying node A with $\theta = 0.1$, the $\sigma$ equals 10% of the concentration of A.

When clicking the submit button, the file and the parameters will be send to the backend server. Once the backend computations are completed, time course data, figures and tables will be send back to the front end.

### B. Server-Based Backend

The backend server is a Linux server where a web server, COPASI and R have been installed. The web server receives the front end requests and sends them to a CGI perl script. The perl script parses the inputs and edits the model file and feeds it to the COPASI to perform the SDE simulations. The simulation results will be saved into files in the tsv (Tab-separated) format. The perl script will feed the result file to the R engine for generating the tables and figures that are HTML ready.

The SDE algorithm first calculates the time step $dt$ first. Then in each step, it calls LSODA to integrate the ODE part and also add the noise part by generating a random value timing square root of $dt$. The random value is generated according to the specified noise types, white or Brownian, relative or fixed.

### C. An Illustrative Example

In the illustrative example, we will develop a simplistic model with three species S (source), I (intermediate) and D (destination), and two reactions, one from S to I and another from I to S. The model is $[S]->[I]->[D]$. The three differential equations are:

$$\begin{cases} \frac{d[S]}{dt} = -0.2[S] \\ \frac{d[I]}{dt} = 0.2[S] - 0.1[I] \\ \frac{d[D]}{dt} = 0.1[I] \end{cases} \tag{4}$$

where [] means the concentration. Their initial concentrations are $[A]_0 = 1.2$, $[B]_0 = 1$, and $[D]_0 = 0.4$ and the unit is $mol/l$. For simplicity, we assume ODE model has been calibrated appropriately. Using the deterministic LSODA algorithm, figure 2 shows the time courses of the species concentrations. The simulated time duration is 10s and simulated in 1000 intervals, i. e., $dt = 0.01s$. It's easily seen that along time [S] decreases, [D] increases, while [I] first increases and then decreases after reaching a peak that is $[I]_{3.45} = 1.20417ml/l$.

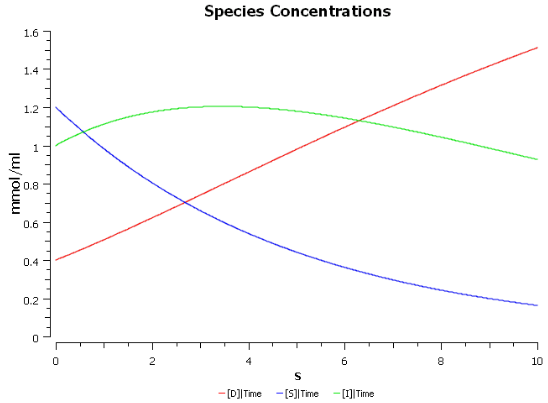


Fig. 2. Time-course concentrations of the three species in ODE model.

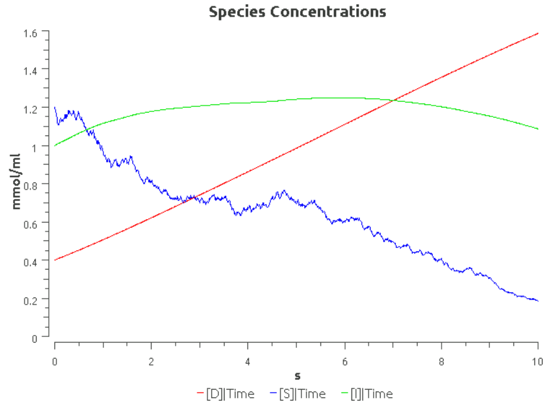


Fig. 3. Time-course concentrations of the three species in SDE model.

Suppose A is a stochastic node, and the noise type is white and relative, and the $\theta$ is 0.1. Now the first ODE equation becomes $\frac{d[S]}{dt} = -0.2[S] + n\sqrt{dt}$, where $n$ is a uniform random variable of mean 0 and standard deviation of $\sigma = 0.1[S]$. Figure 3 is a time course figure of the corresponding species concentrations in the SDE model using the SDE algorithm we implemented. As the stochasticity is introduced into the source node [S], the stochasticity will be propagated into the intermediate and the destination nodes [I] and [D]. The three curves show similar trend as in the deterministic case; but they are stochastic and change in each simulation with different seed of the random number generator. Especially, $[S]$ no longer decreases monotonically but has several local peaks during the simulation.

| | $[S]_{10s}$ | $[I]_{10s}$ | $[D]_{10s}$ | $t_{[I]Peak}$ | $[I]_{Peak}$ |
|---|---|---|---|---|---|
| Determi | 0.1624 | 0.9260 | 1.5116 | 3.4500 | 1.2042 |
| Stoch 1 | 0.1574 | 0.8645 | 1.4501 | 3.1000 | 1.1382 |
| Stoch 2 | 0.0841 | 0.7921 | 1.4588 | 2.6800 | 1.2201 |
| Stoch 3 | 0.1401 | 0.9930 | 1.5541 | 4.5700 | 1.2349 |
| Stoch 4 | 0.1200 | 0.8765 | 1.4783 | 2.9600 | 1.1592 |
| Stoch 5 | 0.2703 | 1.0783 | 1.5997 | 5.2300 | 1.2854 |
| Stoch 6 | 0.1551 | 0.9409 | 1.5121 | 3.0000 | 1.2042 |
| Stoch 7 | 0.1518 | 0.9013 | 1.4844 | 2.9900 | 1.1711 |
| Stoch 8 | 0.1130 | 0.8764 | 1.4946 | 2.8900 | 1.2066 |
| Stoch 9 | 0.1606 | 1.0040 | 1.5731 | 4.1100 | 1.2680 |
| Stoch 10 | 0.1395 | 0.9108 | 1.5080 | 3.9100 | 1.2087 |
| Stoch 11 | 0.1582 | 0.9916 | 1.5582 | 3.8400 | 1.2449 |
| Stoch 12 | 0.1713 | 0.9326 | 1.5028 | 4.0300 | 1.1779 |
| Stoch 13 | 0.2009 | 0.9485 | 1.5103 | 4.3000 | 1.1834 |
| Stoch 14 | 0.1506 | 0.8895 | 1.4803 | 3.3800 | 1.1636 |
| Stoch 15 | 0.1096 | 0.8054 | 1.4383 | 2.7200 | 1.1580 |
| Stoch 16 | 0.1352 | 0.9222 | 1.5113 | 3.0600 | 1.1883 |
| Stoch 17 | 0.2580 | 0.9047 | 1.4553 | 2.1700 | 1.1340 |
| Stoch 18 | 0.1306 | 0.8870 | 1.5008 | 3.6700 | 1.2041 |
| Stoch 19 | 0.0659 | 0.7674 | 1.4186 | 2.7300 | 1.1391 |
| Stoch 20 | 0.2195 | 0.9845 | 1.5524 | 3.4800 | 1.2536 |
| Stoch Ave | 0.1546 | 0.9136 | 1.5021 | 3.4410 | 1.1971 |
| Stoch Std | 0.0513 | 0.0760 | 0.0473 | 0.7560 | 0.0440 |
| Std / Ave | 0.3318 | 0.0831 | 0.0315 | 0.2197 | 0.0368 |

TABLE I
THIS TABLE SHOWS THE END CONCENTRATIONS OF THE THREE SPECIES AND THE PEAK TIME AND VALUE OF THE INTERMEDIATE SPECIES I OF THE DETERMINISTIC AND 20 STOCHASTIC RUNS. THE AVERAGE AND STANDARD DEVIATION ARE ALSO PROVIDED.

We further run the SDE time course for 20 times. Table I shows the species concentrations at the end (10s) and also the peak time and peak value of the intermediate node I. The average values over 20 stochastic runs are close to the values from the deterministic run. The standard deviation of source node concentration [S] at the simulation end is the largest, relative to the average value is about 33%. For [I] and [D], the corresponding percentages are 8.3% and 3.2%, respectively. This confirms that this network is a single direction flow network, from S to D through I. For the peak time, its variation is relatively high, the standard deviation is about 22% of the average peak time. In contrast, the peak value of [I]

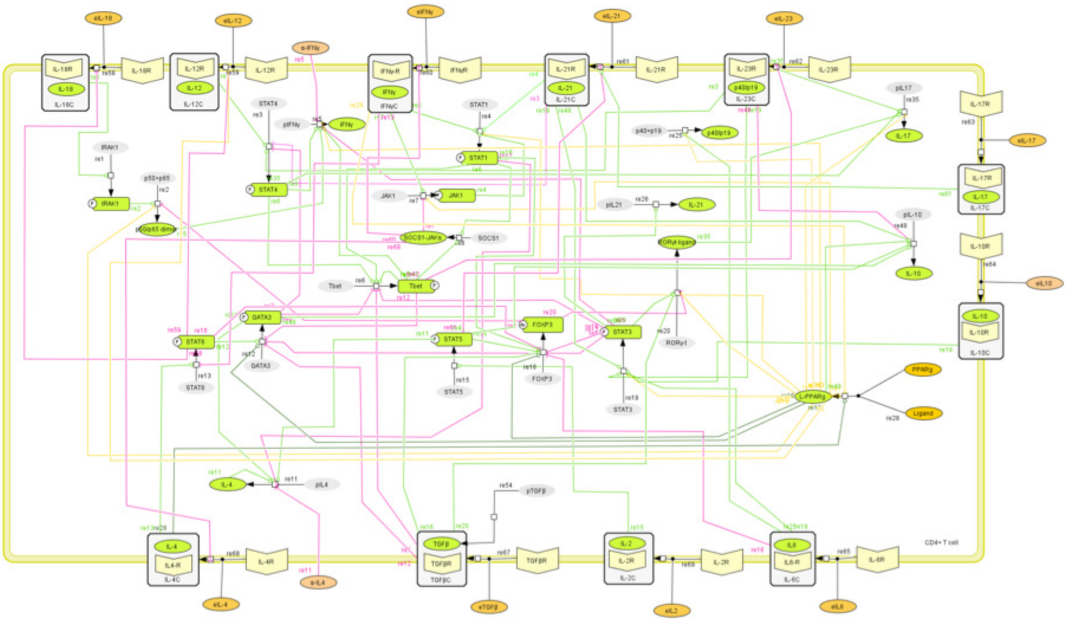


Fig. 4. Systems Biology Markup Language (SBML)-based representation of the CD4+ T cell differentiation model, representing initiation and differentiation fates for T-helper type 1, 2, 17 and regulatory CD4+ T cells (Th1, Th2, Th17 and Treg respectively). For high resolution network model, please refer to www.modelingimmunity.org

has smaller stochasticity.

## V. A Case Study: CD4+ T Cell Differentiation

CD4+ T cells play an important role in regulating acquired immune responses. However they can also contribute to initiating and maintaining pathological responses such as inflammation or autoimmunity. CD4+ T cell differentiation into either effector or regulatory phenotypes is tightly controlled by the extracellular cytokine milieu, complex intracellular signaling networks and specific profiles of transcriptional regulators. Thus, understanding the connections in the intracellular differentiating pathways that control CD4+ T cell homogeneity is key to better understanding mechanisms of regulation and developing novel therapies for infectious diseases, autoimmunity and inflammation.

### *A. CD4+ T Cell Mathematical Model*

To facilitate a comprehensive representation of the dynamics associated with the pathways controlling CD4+ T cell differentiation and plasticity, we constructed an ordinary differential equation (ODE)-based computational model, which includes 93 species, 46 reactions and 60 ODEs driving activations and inhibition pathways (Figure 4). This mathematical model was able to reveal novel unforeseen behaviors as computational hypotheses, which were experimentally validated with immunological in vivo experimentation concerning T-helper type 17 (Th17) and its pro-inflammatory properties, Treg or FOXP3 expressing, and its plasticity triggered by the nuclear receptor PPAR$\gamma$ [6]. However, by being deterministic, this model is partially ignoring random variations in many biological factors, such as transcription and translation rates, and the stochastic nature of the CD4+ T cell differentiation process. Thus, we applied the stochastic modeling technique SDE described here as a tool to explore and analyze how stochasticity in key nodes of the model can explain observed trends in CD4+ T cell differentiation. Moreover, it has been described how the Treg and Th17 phenotypes have a tight equilibrium due to the antagonistic effect of its transcription factors, FOXP3 and ROR$\gamma$t respectively [36] [35]. Thus, SDE modeling can be used to observe how stochasticity can modulate and regulate the steadiness between Th17 and Treg.

### *B. SDE Modeling Results*

The function of the transcription factors ROR$\gamma$t and FOXP3 is tightly regulated by upstream activators STAT3-P and the IL-6 receptor in differentiated Th17 cells. To assess the balance between Th17 and Treg, our CD4+ T cell computational model was induced towards Th17 by adding external IL-6 and TGF-$\beta$. Using the reg-

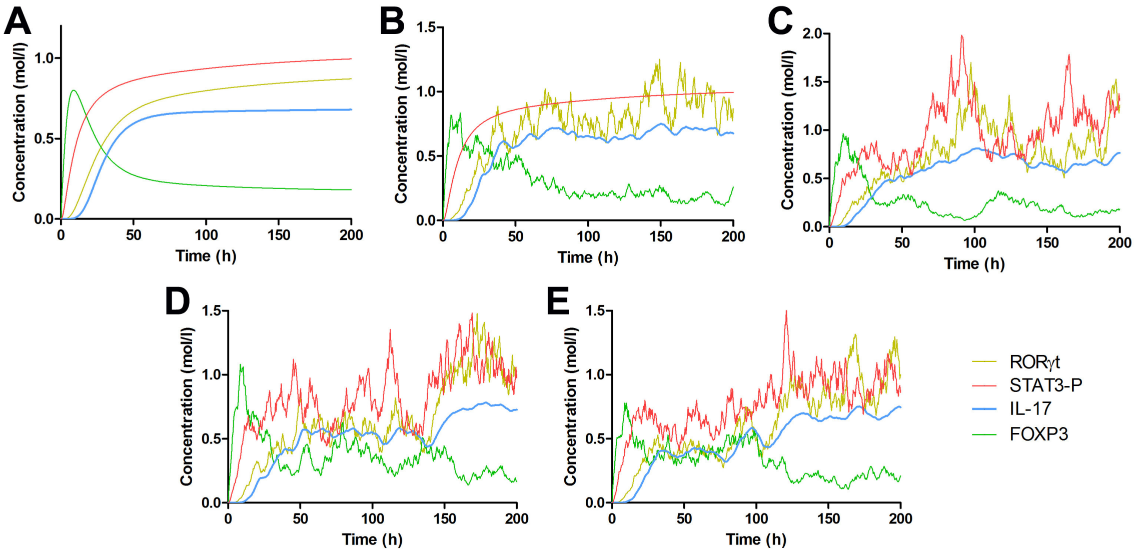


Fig. 5. SDE modeling approaches showed different predictive patters on ROR$\gamma$t-FOXP3 equilibrium when key nodes were stochastically activated. (A) ODE-based time-course performed under Th17 polarizing conditions with IL-6 and TGF-$\beta$ induction. Upregulation on STAT3-P (red) and ROR$\gamma$t (yellow) can be observed together with the down regulation of FOXP3 (green) and a high production of IL-17 (blue). (B) SDE-based simulation in which ROR$\gamma$t and FOXP3 have been added for randomness. (C) SDE-based simulation in which ROR$\gamma$t, FOXP3 and STAT3-P have been added for randomness showing how the production of IL-17 is heavily impaired. (D) SDE-based simulation in which ROR$\gamma$t, FOXP3, STAT3-P and IL-6R (P=0.1) have been added for randomness. (E) SDE-based simulation in which ROR$\gamma$t, FOXP3, STAT3-P and IL-6R (P=0.3) have been added for randomness.

ular ODE approach we could observe an upregulation of Th17-related molecules, STAT3-P and ROR$\gamma$t, as well as an activation of the production of IL-17 (Figure 5A). We observed how the production of IL-17 by Th17 cells in silico was affected by adding stochasticity to the FOXP3 (Treg-related) and ROR$\gamma$t nodes (Figure 5B). We next sought to determine the effect of stochasticity upstream in the Th17-induction pathway. By giving stochasticity to the phosphorylation reaction of STAT3 (activator of ROR$\gamma$) we could observe a more impaired ability for IL-17 to be steadily produced (Figure 5C). This behavior has been observed in our experimental studies, where a non-stable production of IL-17 has been detected when running in vitro induction of Th17 with IL-6 and TGF-$\beta$. We attribute the differences in the percentage of IL-17-producing CD4+ T cells to the variability on STAT3 phosphorylation. To gain a better understanding of this trend, we added stochasticity upstream on the receptor of IL-6. IL-6 is a crucial pro-inflammatory cytokine for Th17 induction. TGF-$\beta$ is a common inductor for both Th17 and Treg [2]. Our results using the SDE tool show a completely broken balance when stochasticity of IL-6 is added (Figures 5D and E). We observed stages where the double-positive, ROR$\gamma$t+ FOXP3+ was generated. Experimentally, the double positive has been already described as a transition between these two phenotypes [33]. Our modeling approaches with the SDE tool predictively show how a double-positive CD4+ T cell can be triggered by adding randomness to the IL-6 binding in the transmembrane IL-6 receptor.

## VI. Conclusions

In this paper, we have made three major contributions. (1) We present SDE and propose that it should be used as a generic stochastic modeling technique for computational biology. (2) We develop a web-based SDE modeling tool that requires minimal effort beyond the effort required for regular deterministic ODE modeling. To our best knowledge, we are the first to build such a user-friendly SDE modeling tool targeting for biologists and immunologists. (3) We demonstrate the SDE modeling technique and our SDE modeling tool with a case study on a CD4+ T Cell Differentiation model that has 93 species and 46 reactions. The case study clearly shows the effectiveness of SDE modeling technique and the power of the SDE tool we have developed.

For future work, we plan to extend our study into the following areas:

- Improvement of the SDE tool and user interface such as adding supporting of multiple runs

- Development of more SDE models based upon biological data through network inference progresses
- Parameter estimation for stochastic processes

## Acknowledgement

This work was supported in part by NIAID Contract No. HHSN272201000056C to JBR and funds from the Nutritional Immunology and Molecular Medicine Laboratory (URL: www.nimml.org).

## References

[1] S. Adra, T. Sun, S. MacNeil, M. Holcombe, and R. Smallwood. Development of a three dimensional multiscale computational model of the human epidermis. *PLoS ONE*, 5(1):e8511, 2010.

[2] E. Bettelli, Y. Carrier, W. Gao, T. Korn, T. B. Strom, M. Oukka, H. L. Weiner, and V. K. Kuchroo. Reciprocal developmental pathways for the generation of pathogenic effector TH17 and regulatory T cells. *Nature*, 441(7090):235–238, May 2006.

[3] E. Borenstein. Computational systems biology and in silico modeling of the human microbiome. *Brief Bioinform*, May 2012.

[4] X. Cai and J. Wen. Efficient exact and K-skip methods for stochastic simulation of coupled chemical reactions. *J Chem Phys*, 131(6):064108, Aug 2009.

[5] Y. Cao, D. T. Gillespie, and L. R. Petzold. The slow-scale stochastic simulation algorithm. *J Chem Phys*, 122(1):14116, Jan 2005.

[6] A. Carbo, R. Hontecillas, B. Kronsteiner, M. Viladomiu, M. Pedragosa, P. Lu, C. W. Philipson, S. Hoops, M. Marathe, S. Eubank, et al. Systems modeling of molecular mechanisms controlling cytokine-driven cd4+ t cell differentiation and phenotype plasticity. *PLoS computational biology*, 9(4):e1003027, 2013.

[7] K. C. Chen, T. Y. Wang, H. H. Tseng, C. Y. Huang, and C. Y. Kao. A stochastic differential equation model for quantifying transcriptional regulatory network in Saccharomyces cerevisiae. *Bioinformatics*, 21(12):2883–2890, Jun 2005.

[8] H. de Jong. Modeling and simulation of genetic regulatory systems: a literature review. *J. Comput. Biol.*, 9(1):67–103, 2002.

[9] C. DeLisi. Mathematical modeling in immunology. *Annu. Rev. Biophys. Bioeng.*, 12:117–138, 1983.

[10] S. Forrest and C. Beauchemin. Computer Immunology. *Immunological Reviews*, 216:176–197, 2007.

[11] D. T. Gillespie. A general method for numerically simulating the stochastic time evolution of coupled chemical reactions. *Journal of Computational Physics*, 22(4):403–434, 1976.

[12] D. T. Gillespie. Exact stochastic simulation of coupled chemical reactions. *The Journal of Physical Chemistry*, 81(25):2340–2361, 1977.

[13] H. J. Gold. *Mathematical modeling of biological systems : an introductory guidebook*. Wiley, New York, 1977.

[14] M. Holcombe, S. Adra, M. Bicak, S. Chin, S. Coakley, A. I. Graham, J. Green, C. Greenough, D. Jackson, M. Kiran, S. MacNeil, A. Maleki-Dizaji, P. McMinn, M. Pogson, R. Poole, E. Qwarnstrom, F. Ratnieks, M. D. Rolfe, R. Smallwood, T. Sun, and D. Worth. Modelling complex biological systems using an agent-based approach. *Integr Biol (Camb)*, 4(1):53–64, Jan 2012.

[15] S. Hoops, S. Sahle, R. Gauges, C. Lee, J. Pahle, N. Simus, M. Singhal, L. Xu, P. Mendes, and U. Kummer. COPASI–a COmplex PAthway SImulator. *Bioinformatics*, 22(24):3067–3074, Dec 2006.

[16] M. Hucka, A. Finney, H. M. Sauro, H. Bolouri, J. C. Doyle, H. Kitano, A. P. Arkin, B. J. Bornstein, D. Bray, A. Cornish-Bowden, A. A. Cuellar, S. Dronov, E. D. Gilles, M. Ginkel, V. Gor, I. I. Goryanin, W. J. Hedley, T. C. Hodgman, J. H. Hofmeyr, P. J. Hunter, N. S. Juty, J. L. Kasberger, A. Kremling, U. Kummer, N. Le Novere, L. M. Loew, D. Lucio, P. Mendes, E. Minch, E. D. Mjolsness, Y. Nakayama, M. R. Nelson, P. F. Nielsen, T. Sakurada, J. C. Schaff, B. E. Shapiro, T. S. Shimizu, H. D. Spence, J. Stelling, K. Takahashi, M. Tomita, J. Wagner, and J. Wang. The systems biology markup language (SBML): a medium for representation and exchange of biochemical network models. *Bioinformatics*, 19(4):524–531, Mar 2003.

[17] S. M. Iacus. *Simulation and Inference for Stochastic Differential Equations: With R Examples (Springer Series in Statistics)*. Springer Publishing Company, Incorporated, 1 edition, 2008.

[18] F. H. Karlsson, I. Nookaew, D. Petranovic, and J. Nielsen. Prospects for systems biology and modeling of the gut microbiome. *Trends Biotechnol.*, 29(6):251–258, Jun 2011.

[19] H. Kitano. Computational systems biology. *Nature*, 420(6912):206–210, Nov 2002.

[20] N. Le Novere and T. S. Shimizu. STOCHSIM: modelling of stochastic biomolecular processes. *Bioinformatics*, 17(6):575–576, Jun 2001.

[21] Z. Liu and Y. Cao. Detailed comparison between StochSim and SSA. *IET Syst Biol*, 2(5):334–341, Sep 2008.

[22] B. D. Macarthur, A. Ma'ayan, and I. R. Lemischka. Systems biology of stem cell fate and cellular reprogramming. *Nat. Rev. Mol. Cell Biol.*, 10(10):672–681, Oct 2009.

[23] B. D. MacArthur, C. P. Please, and R. O. Oreffo. Stochasticity and the molecular mechanisms of induced pluripotency. *PLoS ONE*, 3(8):e3086, 2008.

[24] T. Manninen, M. L. Linne, and K. Ruohonen. Developing It stochastic differential equation models for neuronal signal transduction pathways. *Comput Biol Chem*, 30(4):280–291, Aug 2006.

[25] J. A. Morgan and D. Rhodes. Mathematical modeling of plant metabolic pathways. *Metab. Eng.*, 4(1):80–89, Jan 2002.

[26] B. Novak, Z. Pataki, A. Ciliberto, and J. J. Tyson. Mathematical model of the cell division cycle of fission yeast. *Chaos*, 11(1):277–286, Mar 2001.

[27] B. Novak and J. J. Tyson. Modeling the control of DNA replication in fission yeast. *Proc. Natl. Acad. Sci. U.S.A.*, 94(17):9147–9152, Aug 1997.

[28] L. Petzold and A. Hindmarsh. LSODA (Livermore Solver of Ordinary Differential Equations). *Computing and Mathematics Research Division, Lawrence Livermore National Laboratory, Livermore, CA*, 1997.

[29] R. Ramaswamy and I. F. Sbalzarini. A partial-propensity formulation of the stochastic simulation algorithm for chemical reaction networks with delays. *J Chem Phys*, 134(1):014106, Jan 2011.

[30] A. Saarinen, M. L. Linne, and O. Yli-Harja. Stochastic differential equation model for cerebellar granule cell excitability. *PLoS Comput. Biol.*, 4(2):e1000004, Feb 2008.

[31] E. Schneidman, B. Freedman, and I. Segev. Ion channel stochasticity may be critical in determining the reliability and precision of spike timing. *Neural Comput*, 10(7):1679–1703, Oct 1998.

[32] SDE-Matlab-Toolbox. http://sdetoolbox.sourceforge.net/.

[33] D. M. Tartar, A. M. VanMorlan, X. Wan, F. B. Guloglu, R. Jain, C. L. Haymaker, J. S. Ellis, C. M. Hoeman, J. A. Cascio, M. Dhakal, M. Oukka, and H. Zaghouani. FoxP3+RORgammat+ T helper intermediates display suppressive function against autoimmune diabetes. *J. Immunol.*, 184(7):3377–3385, Apr 2010.

[34] M. Terzer, N. D. Maynard, M. W. Covert, and J. Stelling. Genome-scale metabolic networks. *Wiley Interdiscip Rev Syst Biol Med*, 1(3):285–297, 2009.

[35] X. O. Yang, R. Nurieva, G. J. Martinez, H. S. Kang, Y. Chung, B. P. Pappu, B. Shah, S. H. Chang, K. S. Schluns, S. S. Watowich, X. H. Feng, A. M. Jetten, and C. Dong. Molecular antagonism and plasticity of regulatory and inflammatory T cell programs. *Immunity*, 29(1):44–56, Jul 2008.

[36] L. Zhou, J. E. Lopes, M. M. Chong, I. I. Ivanov, R. Min, G. D. Victora, Y. Shen, J. Du, Y. P. Rubtsov, A. Y. Rudensky, S. F. Ziegler, and D. R. Littman. TGF-beta-induced Foxp3 inhibits T(H)17 cell differentiation by antagonizing RORgammat function. *Nature*, 453(7192):236–240, May 2008.