ON THE STATISTICAL ACCURACY OF STOCHASTIC SIMULATION ALGORITHMS IMPLEMENTED IN DIZZY

Werner Sandmann and Christian Maier

University of Bamberg
Department of Information Systems and Applied Computer Science
Feldkirchenstr. 21, D-96045 Bamberg, Germany
werner.sandmann@uni-bamberg.de

ABSTRACT

Stochastic simulation is in widespread use for analyzing biological pathways. Due to the limited efficiency of a straightforward direct implementation such as the Gillespie algorithm, various improvements and approximate algorithms have been developed. For user-friendliness it is important to have efficient implementations available in software tools. Another important issue is the statistical accuracy of simulation results in terms of variances, confidence intervals, or related measures. We address the problem of computing such statistics for Dizzy, a software tool that has been recommended in a recent study of the user-friendliness of software tools. Therefore, a mathematical framework for statistical output analysis of simulation results is provided, the need for statistics as well as the lack of user support in actually obtaining such statistics with Dizzy and other tools is emphasized, and recommendations for future extensions of software tools are given.

1. INTRODUCTION

In the discrete-state stochastic approach to coupled chemical reactions, the system state is defined by the population of all involved molecular species $S_1, \ldots, S_d$. The time evolution is described by a continuous-time Markov chain $(X(t))_{t\geq 0}$ where $X(t) = (X_1(t), \ldots, X_d(t))$ and $X_i(t)$ is the number of molecules of species S_i present at time t. The Kolmogorov differential equations governing the system dynamics are expressed via the chemical master equation (CME), which is a system of ordinary differential equations (ODEs) where the variables are transient (time-dependent) state probabilities. Since the CME is usually difficult to solve for large or stiff models, stochastic simulation is often applied to analyze biological pathways. Rather than directly solving the CME, realizations of Markov chain trajectories (sample paths) are generated. Stochastically exact trajectory generation is often referred to as the Gillespie algorithm in the context of chemical reactions as Gillespie [2, 3] introduced the terminology of the CME and thereby proposed to use stochastic simulation for system analysis. However, direct simulation where each single reaction is explicitly simulated is exceedingly slow. Therefore, various modified implementations as well as accelerated approximate methods for enhanced trajectory generation have been proposed.

A major drawback of stochastic simulation that has not received much attention in systems biology so far is the statistical uncertainty due to the random nature of simulation results. Despite the fact that Gillespie's algorithm is termed exact, a stochastic simulation can never be exact. The exactness of Gillespie's algorithm is only "in the sense that it takes full account of the fluctuations and correlations" [3] of reactions within a single simulation run. It is common sense in stochastic simulation theory that one should never rely on a single simulation run and Gillespie already mentioned that it is "necessary to make several simulation runs from time 0 to the chosen time t, all identical with each other except for the initialization of the random number generator". In fact, the reliability of simulation results strongly depends on a sufficiently large number of simulation runs, where an explanation of the meaning of a sufficiently large number and the determination of that number has to be carefully done in terms of mathematical statistics. Even with approximate methods for accelerated trajectory generation still a large number of trajectories is required in order to obtain reliable and meaningful results with acceptable statistical accuracy. Hence, in either case stochastic simulation is computationally expensive and can only provide statistical estimates. Mathematically, it constitutes a statistical estimation procedure implying that the results are subject to statistical uncertainty.

An important point is tool support such that stochastic simulation algorithms can be applied by practitioners who need not be experts in stochastic simulation. Recently, Mäkiraatikka et al [5] studied the user-friendliness of software tools and among those studied, all of which had a couple of shortcomings, they recommended Dizzy [6], cf. http://magnet.systemsbiology.net/software/Dizzy.

We address the statistical accuracy of stochastic simulations. When we started our study, the initial intention was to figure out how far accelerated generation of single trajectories comes at the prize of an increased number of trajectories necessary to provide a certain statistical accuracy. This obviously requires an appropriate framework within which this accuracy is measured. It turned out that

currently neither Dizzy nor any of the other tools we are aware of do provide any support with regard to statistical output analysis of simulation results. Consequently, the major focus of our work changed towards introducing an appropriate mathematical framework as well as computing relevant statistical measures. While the mathematical framework is completely tool-independent, we discuss the computation and the further processing of necessary information for statistical measures through Dizzy. To come back to our initial intention we obtained several statistics for two test cases but we did not find any essential differences in the statistical accuracy of the stochastic simulation algorithms implemented in Dizzy. However, this is far from being a general result because the lack of support for statistical analysis and the quasi-manually and thus extremely time-consuming computation of statistics prevented more excessive studies and made it even hard to verify the statistical accuracy for relatively small examples. Though we originally aimed at comparing different algorithms, the statistical accuracy of stochastic simulations is an important property for each algorithm in itself. In fact, it is the only mathematical way to investigate the reliability of simulation results. In practice, the number of simulation runs is usually chosen very large but somewhat arbitrarily. Performing many more runs than necessary for a certain desired statistical accuracy means a significant waste of computer time. On the other hand, too less simulation runs render the results meaningless. Hence, it is highly desirable to have some rules giving the required number of simulation runs. In particular, we strongly emphasize the urgent need for integrating statistical output analysis into Dizzy and other tools in a user friendly way and we provide hints and recommendations how this should be done.

The remainder of this paper is organized as follows. In Section 2 we outline how simulation outcomes can be formalized in a unified way such that they yield to statistical analysis. Measures for the statistical analysis are given in Section 3. Then we present our test cases and briefly describe how we computed statistics from the results provided by Dizzy. Finally, we give conclusions and recommendations for future tool extensions.

2. FORMALIZING SIMULATION OUTCOMES

Stochastic simulations are nothing else than statistical estimations using computers. They generate realizations of random variables with the help of random number generators. Similarly as for observations from laboratory experiments, several properties can be derived from the realizations. Thus, from a statistical point of view repeated laboratory experiments and stochastic simulations are equivalent. The only difference is in the way realizations are generated. In a laboratory experiment they are generated within a physical real life environment whereas a stochastic simulation imitates real life environments by using appropriate rules.

In practice, each simulation run is finished at some time and the outcome is a finite sequence of states where state changes are triggered by reactions and several properties can be immediately derived for all species. Such properties can be mathematically described as a function f of the sequence of states. Since outcomes of stochastic simulations are realizations of random variables and functions of random variables are again random variables, the property of interest is also a random variable. We denote it by $Y = f(X(t_0), \ldots, X(t_m))$. Note that although the set of reaction times is countable, yielding a sequence of states, the time differences are in general not equal, i.e. typically $t_{i+1} - t_i \neq t_{j+1} - t_j$ for $i \neq j$. The random variable Y may be the number of molecules of a species at some (not necessarily reaction) time t in which case it is simply the projection to the relevant component of $X(t)$. It may also be the mean number of molecules, the time until a specific number of molecules has been reached or exhausted. In general, Y might be any imageable property that can be determined from a sample path. Each time a realization is generated, it is different in general. Also it will rarely ever exactly coincide with the "true" value Y. Statistical methods are required to assure that no wrong conclusions are drawn from accidentally untypical experiments. More precisely, a statistical estimation procedure must be executed up to some predefined accuracy.

According to classical statistics one builds an *estimator* from several (say N) stochastically independent and identically distributed (iid) random variables, generates N realizations via experiments, and estimates the property of interest by the resulting realization of the estimator. Since an estimator is itself a random variable it follows a probability distribution with mean (expectation), variance, higher moments etc. Hence, it is important to know its fluctuation. The characteristics of the estimator, in particular its variance and measures derived from it, determine the accuracy and the reliability of the estimate.

3. STATISTICAL ACCURACY OF SIMULATIONS

In this section we elaborate on the statistical estimation procedure which is needed and performed in stochastic simulations thereby focusing on the expectation $E[Y]$. We particularly emphasize the large time complexity and the nevertheless remaining inherent uncertainty.

3.1. Point Estimators and Confidence Intervals

Given a sample $Y_1, \ldots, Y_N$, independent and identically distributed as a univariate random variable Y, the natural estimator for $E[Y]$ is the *sample mean*

$$\bar{Y} = \frac{1}{N} \sum_{i=1}^{N} Y_i. \tag{1}$$

It is important to note that the sample mean is an *unbiased* estimator for $E[Y]$, i.e. $E[\bar{Y}] = E[Y]$. Unbiasedness of an estimator is an obviously desirable property, but for more complicated properties than the expectation often not so straightforward to obtain as it might appear. As a simple example note that an unbiased estimator for

the variance $\sigma^2(Y)$ is given by

$$S^2 = \frac{1}{N-1}\sum_{i=1}^{N}\left(Y_i - \bar{Y}\right)^2, \qquad (2)$$

whereas the probably first suggestion to divide the sum by N instead of $N-1$ yields a biased estimator.

As a random variable, an estimator is subject to statistical uncertainty, and the question arising after an estimator has been chosen is that of accuracy or reliability in a statistical sense. Unbiasedness is not a sufficient criterion to assure satisfiable accuracy. In addition the estimator's variance is of major importance. In fact, what is needed to make proper statements on the accuracy, in particular dependent on N, is a *confidence interval*.

A confidence interval is a random (dependent on the random sample) interval that contains the property of interest with some predefined probability $1-\alpha$, where $1-\alpha$ is called the *confidence level*, which is in practice usually chosen as 90%, 95% or 99%. According to the central limit theorem, for sufficiently large N classical statistics gives us the confidence interval

$$C = \left[\bar{Y} - z_{1-\alpha/2}\sqrt{\frac{S^2}{N}}\ ,\ \bar{Y} + z_{1-\alpha/2}\sqrt{\frac{S^2}{N}}\right] \qquad (3)$$

where $z_{1-\alpha/2}$ denotes the $1-\alpha/2$ quantile of the standard normal distribution. An important point is how to interpret confidence intervals. As explained above, experiments generate realizations of all random variables involved in the estimation procedure yielding specific values called estimates. In particular, for a given set of realizations $y_1, \ldots, y_N$ one gets a realization of the confidence interval where endpoints are numerical values and the confidence interval realization either contains $E[Y]$ or not. Thus, there is nothing probabilistic *after* the realizations have been obtained and the endpoints have been accordingly set to numerical values. It is a wrong interpretation that each single confidence interval realization contains $E[Y]$ with probability $1-\alpha$. The correct interpretation is that if one constructs a large number of $100\cdot(1-\alpha)$% confidence interval realizations, each based on N experiments, the proportion (*coverage*) of those that contain (cover) $E[Y]$ is $1-\alpha$. A direct consequence of the correct interpretation of confidence intervals is that one might obtain confidence interval realizations that do not contain $E[Y]$ at all.

3.2. Required Number of Simulation Runs

The width of the confidence interval suggests the amount of variability in the estimated value. As the interval is symmetric meaning that $\bar{Y}$ is the midpoint, it is sufficient to consider the confidence interval half width. In non-simulative computations the relative error is most often more meaningful than the absolute error. Similarly, the relative half width of the confidence interval is an appropriate measure of simulation accuracy.

In iterative numerical computations one proceeds by iterating up to a given accuracy, more specifically up to a maximum relative error. Analogously, a stochastic simulation can be viewed as a kind of iteration where simulation runs must be generated until the accuracy is sufficient which means until the relative confidence interval half width for a given confidence level is less than a given maximum error bound. Obviously, the number of required simulation runs is not fixed in advance since the realizations of the confidence interval depend on the specific outcomes of the simulation runs. As an expression for the number of simulation runs required to meet a predefined maximum relative error of β and a confidence level of $1-\alpha$ expression (3) yields

$$N \geq \frac{z_{1-\alpha/2}^2 S^2}{\beta^2 \bar{Y}^2} = \frac{z_{1-\alpha/2}^2}{\beta^2}\cdot\frac{S^2}{\bar{Y}^2}. \qquad (4)$$

Since S^2 and $\bar{Y}$ are estimators for the variance and the expectation, respectively, the ratio $S^2/\bar{Y}^2$ is an estimator for $c_Y^2 = \sigma^2(Y)/E[Y]^2$, the squared *coefficient of variation* of Y which is sometimes also called the (estimated) *relative error of the estimator* $\bar{Y}$.

Now, we can put specific values for the confidence level and the maximum relative error into expression (4). Taking usual values such as a confidence level of 99% and a maximum relative error of 10% we get $z_{1-\alpha/2} \approx 2.58, \beta = 0.1$, and thus $N \geq 664\cdot c_Y^2$. As we can see N is determined by the squared coefficient of variation which is the reason that in some cases simulation can be very proper whereas in other cases it results either in runtime explosion or unsatisfactory inaccuracy. More precisely, if c_Y^2 is close enough to zero, a moderate number of simulation runs suffice but if c_Y^2 is large, the required amount of simulation runs grows enormously. As an extreme example take a situation where a very small probability γ of some event has to be estimated. Such a probability can be estimated via the expectation of the event's indicator function. Then $c_Y^2 = (1-\gamma)/\gamma$ is extremely large for very small γ. To be more specific, with the accuracy requirements stated above the required number of simulation runs in (4) to estimate a probability of 10^{-9} is $N \geq 6.64\cdot 10^{11}$.

Although the latter example might seem unrealistically at a first glance, there are in fact a lot of situations where exactly this problem occurs. Even if we are not concerned with such extreme cases it must be noted that except for cases where the squared coefficient of variation of the property of interest is close to zero, simulation requires a large amount of computer time and at least as seriously there remains a non-negligible probability of getting a wrong estimate.

4. OBTAINING STATISTICS IN DIZZY

The stochastic simulation algorithms available in Dizzy are Gillespie's direct method [2, 3], the so-called Gibson-Bruck algorithm [1] which is an implementation of an equivalent interpretation of the Markov chain dynamics, and two versions of tau-leaping [4], an approximate multistep approach for accelerated trajectory generation.

Dizzy provides a graphical user interface as well as a command-line interface. Unfortunately, neither of these

interfaces provides any support for statistical analysis. The only related option is to compute steady state fluctuations but with regard to potentially infinite time horizons within a trajectory. We computed all previously introduced statistical measures manually for various parameter sets of two test-cases. The first one, the enzymatic reaction set

$$E + S \underset{c_2}{\overset{c_1}{\rightleftharpoons}} ES \xrightarrow{c_3} E + P \tag{5}$$

is one of the small examples that comes with Dizzy. The second one is a part of the bacteriophage λ pathway, the lysis-lysogeny switch whose reaction kinetics are given in Table 1. As mentioned in the introduction, for these reaction sets we did not find any significant differences in the statistical accuracy of the stochastic simulation algorithms implemented in Dizzy. It does not make much sense to present excessive tables in order to illustrate this. So, also due to lack of space we omit it.

Table 1. Lysis-lysogeny switch in bacteriophage λ

$2X$	$\overset{c_1}{\rightleftharpoons}$	X_2	dimerization
$D + X_2$	$\overset{c_2}{\rightleftharpoons}$	DX_2	binding 1)
$D + X_2$	$\overset{c_3}{\rightleftharpoons}$	DX_2^*	binding 2)
$DX_2 + X_2$	$\overset{c_4}{\rightleftharpoons}$	DX_2X_2	binding 3)
$DX_2^* + X_2$	$\overset{c_5}{\rightleftharpoons}$	DX_2X_2	binding 3)
D	$\xrightarrow{c_s}$	$D + X$	slow transcription
X	$\xrightarrow{c_d}$	$\emptyset$	degradation
DX_2	$\xrightarrow{c_f}$	$DX_2 + X$	enhanced transcription

We were restricted to these rather small examples because all statistics had to be essentially computed manually. Though Dizzy offers the opportunity for performing many independent simulation runs specified as the ensemble size, it does not provide all "subresults" for each run. Three output options are available. The plot option yields, as the name suggests, a plot of the numbers of molecules versus time but gives no numerical values. The other options are tables and their storage where the number of intermediate time points can be specified but for each time point only mean values of molecular numbers averaged over the simulation runs are provided. That is, only sample means are computed without variances, etc. Therefore, we obtained the necessary information for each simulation run one after another. More precisely, for each configuration we performed N single simulation runs by invoking the chosen simulation algorithm N times by hand. The reader may imagine the enormous amount of time wasted. In fact, this way the simulation became interactive in that each simulation run had to be started manually. Fortunately, Dizzy uses fresh random number also when single runs are manually performed one after another and not only when many independent runs are performed automatically. Finally, we proceeded by transferring the outcomes of each run to a statistical software package (S-PLUS) which provided us with the desired statistical measures.

5. CONCLUSIONS AND RECOMMENDATIONS

The statistical accuracy of stochastic simulations is an important but so far largely neglected issue in order to measure the reliability of simulation results. A mathematical framework for unified statistical simulation output analysis can be given by appropriately formalizing simulation outcomes and handling the property of interest, formally expressed as a function of random variables which is itself a random variable, by means of classical statistics. User support for statistical analysis is lacking in current software tools for simulating biological pathways. As statistical accuracy is essential for meaningful results, such a user support is highly desirable and strongly recommended. Hence, future extensions of software tools should integrate the methods outlined here. It seems that this should not be too difficult to implement and rather straightforward if the property of interest is related to the number of molecules at one or more specific times. In such cases, all required information is actually computed within a stochastic simulation and it remains to appropriately process it and provide it to the user. Another recommended feature is to offer the user the opportunity to prespecify the desired statistical accuracy, e.g. in terms of relative errors or relative confidence interval half-width, and automatically perform simulation runs until this accuracy is reached. It would be also of interest to provide a more flexible specification of the time horizon for each simulation run. Properties of practical interest are times until the molecules of certain species are exhausted or certain subsets of the state space are reached. Accordingly, users should be allowed to specify such terminating conditions for simulation runs.

6. REFERENCES

[1] M. A. Gibson and J. Bruck. Efficient exact stochastic simulation of chemical systems with many species and many channels. *J. Phys. Chem.*, 104:1876–1889, 2000.

[2] D. T. Gillespie. A general method for numerically simulating the time evolution of coupled chemical reactions. *J. Comp. Phys.*, 22:403–434, 1976.

[3] D. T. Gillespie. Exact stochastic simulation of coupled chemical reactions. *J. Phys. Chem.*, 71(25):2340–2361, 1977.

[4] D. T. Gillespie and L. R. Petzold. Improved leap-size selection for accelerated stochastic simulation. *J. Chem. Phys.*, 119:8229–8234, 2003.

[5] E. Mäkiraatikka et al. Stochastic simulation tools for cell signaling: survey, evaluation and quantitative analysis. In *Proc. 2nd Conf. Foundations of Systems Biology in Engineering*, pages 171–176, 2007.

[6] S. Ramsey, D. Orrell, and H. Boulouri. Dizzy: Stochastic simulation of large-scale genetic regulatory networks. *Journal of Bioinformatics and Computational Biology*, 3(2):415–436, 2005.