RSC Advances

ROYAL SOCIETY OF CHEMISTRY

www.rsc.org/advances

The Accounting of Noise to Solve the Problem of Negative Populations in Approximate Accelerated Stochastic Simulations

Shantanu Kadam[†] and Kumar Vanka[†*]

[†]*Physical Chemistry Division, National Chemical Laboratory, Dr. Homi Bhabha Road,*

Pashan, Pune, Maharashtra – 411 008, India

*Corresponding author. E-mail: k.vanka@ncl.res.in

Abstract:

The advent of different approximate accelerated stochastic simulation methods have helped considerably in reducing the computational load of the exact simulation algorithms. However, along with the reduction in the computational load comes the risk of driving the molecular numbers to the regime of negative numbers during the simulations. Over the years, various methods have been developed in order to solve the problem by using different strategies. Some methods have employed binomial numbers to model the reactions, while others have tried the partitioning of the reaction network. In this manuscript, we have proposed a new approach where the noise inherent in the choice of the number of firings of a given reaction during a time step is taken into account. This idea of noise accounting is used in conjunction with the accelerated stochastic method: the Representative Reaction Approach (RRA). It is found that the new method is successful at solving the problem of negative numbers, and compares very favorably with other state-of-the-art stochastic simulation methods.

Introduction

The stochastic time evolution of a chemical system can be described by the Chemical Master Equation[1,2] (CME). But owing to its complexity, solving the CME is a difficult task and one has to rely on Monte Carlo simulation techniques that generate stochastic realizations of the underlying chemical kinetics. One such technique is the kinetic Monte Carlo[3] based Stochastic Simulation Algorithm[4,5] (SSA) developed by Daniel Gillespie. This technique simulates a randomly chosen *single* reaction during each time step giving stochastic realizations until a desired time is reached. However, this approach is demanding for the simulations of realistic systems. Subsequent to the development of the SSA, several methods have been developed in order to improve the performance of the SSA, such as the next reaction method[6], the optimized direct method[7] , the sorting direct method[8] and the more recent recycling direct method[9] (RDM). In addition to these methods, the Delay Stochastic Simulation Algorithm[10] (DSSA), which considers time delays, has also been developed. It has been found that such attempts to increase the computational performance of the SSA have only been marginally successful.

The lack of significant success in improving the SSA with exact simulation approaches has led to the development of new approximate methods, where some of the accuracy of the SSA has been sacrificed. One such approach consists of hybrid methods[11-13], which have been used for the multiscale simulations of chemical systems. In these methods, the Chemical Langevin Equation[14] or the Reaction Rate Equations are coupled with the SSA. Even though the hybrid methods have succeeded in reducing the computational load of the SSA to some extent, they have lost the simplicity of the SSA. Another approach consists of leaping methods, where larger time steps are taken in order to simulate the occurrence of more reactions. In one such method,

which was the first of its kind, Daniel Gillespie proposed Gillespie's Approximate Stochastic Algorithm[15] (GASA). In this method, the time step during the simulations is derived from the Leap Condition[15]: a condition wherein a change in the number of reactant molecules in a given reaction is allowed as long as it alters the "propensity function" (the product of the reactant number of molecules and the rate constant) by an infinitesimal amount for that reaction. This method has helped to reduce the computational load of the simulations. Over the years, several improvements to this approach have been proposed, which includes the Gillespie-Petzold[16] (G-P) method, the implicit tau-leaping method of Rathinam *et al*.[17], the efficient step size method of Cao *et al*.[18], the *K*-leap method of Cai and Xu[19], the N-leap method of Xu and Lan[20] and the recent Representative Reaction Approach[21] (RRA) that we have developed.

In all the approximate accelerated methods mentioned above, the reaction numbers are modeled by a Poisson distribution. Since the range of random variables generated by the Poisson distribution is unlimited, some reactions will fire many more times, thereby giving rise to physically unrealistic or negative numbers during the simulations. In other words, the occurrence of the negative population during the simulations can also be interpreted as the consequence of a violation of the Leap Condition. One obvious way to avoid the occurrence of negative populations is to model reaction numbers by random variables that have a finite range. Hence, simulation methods which use binomial random variables were developed. They include the BD - τ leap methods of Tian-Burrage[22], Chatterjee *et al*.[23], the multinomial τ - leap[24] approach, the efficient binomial leap[25], the R-leap[26], and the Generalized binomial leap[27] for delayed reactions, as well as the RRA used in conjunction with the binomial distribution.[28] Apart from these methods, Cao *et al*.[29] have developed a method where the reaction network is partitioned

into critical and noncritical reactions. This same concept of partitioning of a reaction network has been used by Yates *et al*.[30] in their confidence-based method. In case of such methods, the noncritical reactions were modeled by Poisson variables.

However, all the methods mentioned in the above paragraph have their own pros and cons. The BD-τ leap method of Tian-Burrage[22] is based on the concept of limiting reactants, which is used to determine the upper bound on the maximum allowed firings of each reaction channel during a leap. This constraint seems to be artificial in some situations; for instance, in reaction networks in which there are certain reactions that tend to increase the consumed reactant numbers for that particular reaction. Along with this, this method also fails to simulate the cases where there are multiple channel reactant dependencies, i.e. cases where a single reactant gets consumed in multiple reactions. Chatterjee *et al*.[23] approached this problem by employing the binomial distribution while updating the currently available molecular population. This introduces some bias in the choice of the reaction numbers, as the earlier reactions occur more frequently than those selected afterwards. In other words, it depends on the order in which the reactions are selected. A potential solution to this problem that has been suggested is to choose the reactions randomly at each time step. The efficient binomial leap attempts to solve the problem, but it becomes slow due to the requirement of some more binomial random variables at each time step. Cao *et al*.[29] tried to fix the problem, but their solution became less flexible with the introduction of the second control parameter. Along with this, the use of Poisson random variables keeps alive the risk of physically unrealistic numbers being obtained. The multinomial τ-leaping[24] and R-leaping,[26] which are extensions of the binomial methods, have obtained

some success at solving the aforementioned problems, but lose the computational simplicity of these methods.

Therefore, as the paragraphs above indicate, there are difficulties inherent in all the previous methodologies that have been employed to date. In this current work, we report a new approach that we have adopted in order to try and surmount these difficulties. This new method is primarily based on the notion of "noise", and works in conjunction with the Representative Reaction Approach (RRA). The new approach is based on the reasoning that approximate stochastic simulation methods fail because the number of firings determined for each reaction during the simulation step by these methods is greater than the appropriate number. In other words, if "*y*" is the number of firings determined by an approximate accelerated method for a given reaction during a given time step, it is in excess of the appropriate value, say "*x*", by a value "*n*". By "appropriate", what is meant is that "*x*" is the value of the number of firings of that specific reaction that would be perfect in keeping with the Leap Condition. The implication here is that if "*x*" had been chosen as the number of firings by the approximate accelerated method, instead of "*y*", then the simulation would have proceeded perfectly, without encountering problems such as that of negative populations. So, if one could find the correct number of firings ("*x*") for each reaction in each time step, one could proceed with the accelerated simulation. Now, since $y = x + n$, if one could determine the amount (n) by which y exceeds the appropriate number of firings, x, then one could determine x and thus proceed with an accelerated algorithm that would give results faster than SSA but with problems such as negative populations eliminated. But how would one find n ?

We postulate that n is the noise inherent in the determination of the number of firings (y) for a given reaction by the approximate accelerated method. Now, "noise" in this context in the system can have both positive and negative values, because actual stochastic dynamics can both slow down or accelerate at any stage of the reaction. However, for the leap in question, where the number of firings has led to negative populations, the noise correction can *only be a subtraction* from the determined number of firings of the given reaction. This is because considering the fluctuations/noise as a positive correction to the number of firings would lead to even more unphysical negative values for the populations. Therefore, such corrections, while they can be calculated, are discarded.

Hence, if the noise n is subtracted from y, viz. $x = y - n$, then one would obtain the correct number of firings for each step, in accordance with the Leap Condition. This is further illustrated in Figure 1 below.

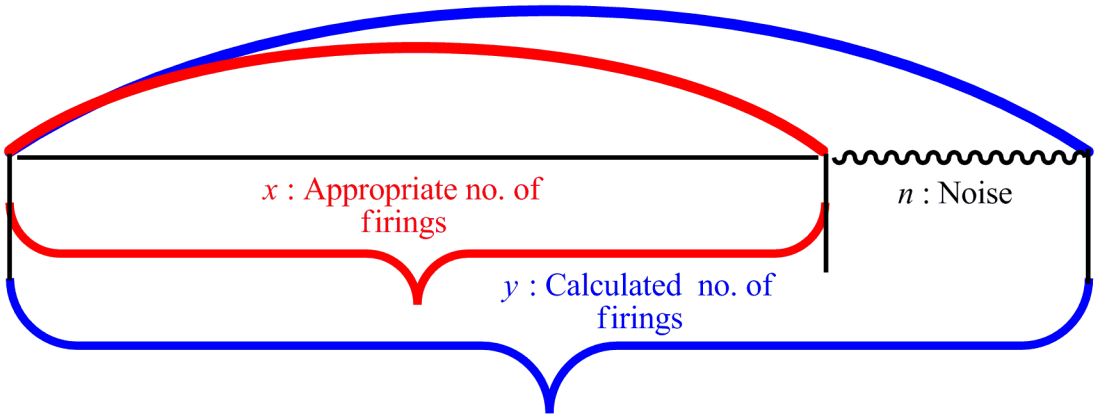

Figure 1. The pictorial theme of the concept of noise associated with every reaction; in a particular time step, x is the appropriate number of reactions, y is the calculated number of reactions and n is the associated noise.

We have tested this idea for the case of the Representative Reaction approach (RRA), an accelerated stochastic method that we have developed[21], by incorporating this new concept of subtracting the noise, *n*, from the number of firings obtained for each step for every reaction. This new approach (termed as RRA-Noise), has been compared to a number of other accelerated methods that have been proposed in the literature, including GASA[15], G-P[16], the BD-τ of Chatterjee *et al.* [23] etc. The current approach provides results which compares very favorably with other approaches, in addition to being relatively simple and easy to implement.

The rest of the paper is organized as follows: in the Methodology section, we have discussed in brief the necessary background required for the theoretical discussion, with the description of the new method followed by the implementation details of the same. In the Results and Discussion section, simulations of different examples have been reported that confirm the reliability and efficiency of the newly proposed approach. The conclusions are provided in the last section.

Methodology

Background

A well-stirred mixture of *N* chemical species $\{S_1, S_2, ..., S_N\}$, which are interacting with each other through *M* chemical reactions $\{R_1, R_2, ..., R_M\}$ has been considered. The mixture is assumed to be in thermal equilibrium at some finite temperature T. The state of this mixture at any particular time, *t*, is specified by a state change vector: $X(t) \equiv (X_1(t), X_2(t), ..., X_N(t))$. Our aim is to study the time evolution of this *N* component vector from some given initial conditions,

say, $X(t_0) \equiv x_0$. Each chemical reaction R_j in the mixture is characterized by a propensity function, a_j and by the state change vector, $\nu_j = (\nu_{1j}, \nu_{2j}, ..., \nu_{Nj})$. Here, ν_{ij} is the change produced by the R_j reaction in the molecular population of the S_i species. The quantity $a_j(x)dt$ gives the probability that the R_j th reaction will occur somewhere in the next infinitesimal time interval $[t, t+dt)$.

Concept of Noise

As mentioned in the Introduction, the current approach is to determine the value of the number of firings (*n*) that is in excess of the appropriate value (*x*) that would be in accordance with the Leap Condition. This value *n* is determined as the noise present in the number of firings (*y*) calculated by the approximate accelerated stochastic method. In order to determine the value of the noise, *n*, we calculate the "Poisson noise" for every reaction firing value calculated for every step.

In an attempt to accelerate the SSA, Gillespie had modeled[15] the occurrences of different chemical reactions by Poisson random variables. Since the number of events (chemical reactions) taking place in a specific time interval are discrete in nature, it is apt to model them by the Poisson probability distribution. In other words, the firings of chemical reactions are treated as Poisson processes. It was further shown by Gillespie that the mean (or expected) value and the variance of the R_j th reaction is $a_j\tau$. In a Poisson process, the actual number of reactions fluctuates about its mean value, $a_j\tau$, with a standard deviation of $\sqrt{a_j\tau}$. These fluctuations in

the reaction numbers are treated as Poisson noise. In electronics, similar fluctuations are known as "shot noise".[31,32]

The fluctuation in every individual reaction implies that all reactions in the chemical system are accompanied by noise. The noise also expresses the basic form of uncertainty associated with the occurrence of the reactions. The uncertainty is substantial when the number of molecules participating in such reactions (or the propensity function) is small enough. It can be negligible (or very small), when the number of molecules (or the propensity function) are abundant. This means that the reactions in any chemical system are always accompanied by the noise. The strength of the noise associated with a reaction varies as the square root of the expected number of firings of the given reaction. Thus, the noise relatively decreases as the expected number of firings of the reaction increases. However, the ratio of expected number of reactions to the noise, i.e., $\frac{a_j \tau}{\sqrt{a_j \tau}}$, increases.

In case of chemical systems that have less number of molecules, the simulations may show unfeasible fluctuations, which, in turn, may give rise to unrealistic (or negative) numbers. Hence, the occurrence of negative numbers can be avoided by removing such unfeasible fluctuations that are in the form of noise. Furthermore, it will be shown in the Results and Discussion section that the removal of noise associated with every reaction does not affect the accuracy of the simulations.

Representative Reaction Approach (RRA) with Noise

In order to speed up the SSA simulations, the Representative Reaction Approach[21] (RRA) has been proposed. In this recently proposed method, the chemical system to be simulated is represented by a single representative reaction (RR). The reaction that has been found to be the most effective is $2A \rightarrow B$. Like any other reaction, the RR is also characterized by the rate constant and the propensity function. The propensity function, $a_0(x)$ is the sum of propensities of all the individual reactions and the rate constant, C_0 is a weighted average of all the rate constants. Thereafter, the total number of hypothetical species, x_0 are calculated. Furthermore, by applying the leap condition to this RR, the expected reactions that are supposed to take place in the next time step are determined. The firings of the individual reactions are modeled by using the Poisson random number generator.[33] However, as discussed in the Introduction, such approximate accelerated methods as the RRA are prone to exhibiting negative numbers during the simulations, for certain reaction systems.

The current approach is to employ the notion of noise whenever negative numbers are encountered in a given step during the simulation. Initially, the simulations of any chemical system are carried out in the usual way by the RRA approach. When negative numbers are obtained at any time step, that step of the RRA is annulled. Working on the assumption that the negative numbers obtained are an indication of excess noise in the leap (see Figure 1), the current method attempts to reduce the uncertainty in the fluctuations by removing the noise from the expected number of reactions. The procedure for the noise-elimination based approach that is employed along with the RRA, is provided in the next subsection.

Steps for the Implementation of RRA-Noise

The implementation details of the new method, RRA-Noise, are outlined below:

Step 1: input the initial number of species and the rate constants of the constituent reactions; initialize the counters and the random number generators to a seed value and transfer the initial number of species to some temporary locations (variables).

Step 2: calculate the propensity functions: $\{a_1, a_2, ..., a_M\}$

the sum of the propensity functions : $a_0(x) = \sum_{j=1}^{M} a_M$

the weighted rate constant: $C_0 = \sum_{j=1}^{M} \left(\frac{a_j(x)}{a_0(x)} \right) c_j$

Step 3: calculate the total number of species present: $x_0 = \frac{C_0 + \sqrt{C_0^{\,2} + 8a_0c_0}}{2c_0}$

Step 4: calculate the time step $\tau = \frac{N_0}{a_0(x)}$, where the total number of reactions are:

$N_0 = \frac{16\varepsilon a_0(x)}{C_0(2x_0 - 1)}$, the value of ε being 0.06.

Step 5: calculate the expected number of reactions for the individual reactions: $\exp_j = a_j\tau$

Step 6: calculate the actual number of firings of individual reactions: $k_j = poidev(\exp_j, iseed)$

Step 7: make the necessary changes in the molecular populations using the appropriate stoichiometric parameters and reaction numbers.

Step 8: if negative numbers are not found, continue with the RRA; else discard the step and use the initial species stored for that step in the temporary locations.

Step 9: calculate the noise: $\sigma_j = \sqrt{a_j \tau}$

Step 10: calculate the corrected expected number of reactions: $\exp'_j = \exp_j - \sigma_j$

Step 11: calculate the new actual number of reactions: $n_j = poidev(\exp'_j, iseed)$

Step 12: make the necessary changes in the molecular populations.

Step 13: go to step 1.

Results and Discussion

Discussed below are the results of simulations done for four different chemical systems. In addition to the simulations done with the newly proposed RRA-Noise method, simulations have also been done with the Stochastic Simulation Algorithm[4,5] (SSA) and the approximate accelerated methods: the Gillespie's Approximate Stochastic Algorithm (GASA)[15], the Gillespie- Petzold (GP) method[16] and the Binomial distribution based tau (BD-τ) method of Chatterjee *et al*.[23]. This section discusses the results of the simulations for the different systems and provides a comparison of the efficiency and robustness of the RRA-Noise method in comparison to the other methods. Specifically, what was compared was (i) the means and the coefficient of variations (CVs) obtained for 500 simulation runs for each method, and (ii) the average CPU times and the number of steps for a simulation obtained from the CPU times of the

500 simulation runs for each method. This was done for all the five chemical system examples considered. The stability analysis of the newly proposed algorithm has been discussed for the system of first order reactions and the oscillatory reaction model.

The Carletti-Burrage Model

The following reaction network model was proposed by Carletti and Burrage.[25]

$$
\begin{aligned}
&R_1 : RNA \xrightarrow{c_1} DNA1 \\
&R_2 : DNA1 \xrightarrow{c_2} RNA \\
&R_3 : m \xrightarrow{c_3} RNA \\
&R_4 : RNA \xrightarrow{c_4} m \\
&R_5 : 2m \xrightarrow{c_5} D \\
&R_6 : D \xrightarrow{c_6} 2m \\
&R_7 : DNA + D \xrightarrow{c_7} DNA1 \\
&R_8 : DNA1 \xrightarrow{c_8} DNA + D \\
&R_9 : DNA1 + D \xrightarrow{c_9} DNA2 \\
&R_{10} : DNA2 \xrightarrow{c_{10}} DNA1 + D
\end{aligned}
\qquad (1)
$$

where RNA, DNA, DNA1, DNA2, D and m are the species taking part in the different reactions; and the symbols (c_1 to c_{10}) over the arrows indicate the rate constants of the respective reactions.

The numerical values of the rate constants of the reactions and initial molecular species are given in Table S8 and Table S9 of the Supporting Information.

The Carletti-Burrage Model is simulated by different methods which have been discussed earlier in the manuscript. In case of simulation by Gillespie's Approximate Stochastic Algorithm

(GASA), we found that negative molecular numbers occurred for some species during certain steps of the simulations. Thus, GASA was found to be unsuitable for the simulation of this model. Moreover, the binomial distribution based tau (BD-τ) method of Tian-Burrage[22] could not be applied for this model, since there are some species which take part in multiple reactions: a situation that the BD-τ method of Tian-Burrage is incapable of handling, making it technically non-applicable for such reaction networks. Hence, only the methods that were successfully able to reproduce the simulation trajectories are reported here. They are: the Stochastic Simulation Algorithm[4,5] (SSA) , the Gillespie-Petzold (G-P) method[16], the binomial distribution based tau (BD-τ) method of Chatterjee *et al*.[23], and our newly proposed method: RRA-Noise.

The error control parameter, ε, with a standard value of 0.03 was used for doing the simulations with GASA and G-P (this value of ε has been the standard value employed in previous reports[15,16]), while, in the case of the RRA-Noise, the value of ε has been taken as 0.06. These ε values have been used for all the subsequent examples of the chemical systems simulated by these methods. As mentioned earlier, the values of the means (with their respective error bars) and the CVs reported in Figure 2 and Figure 3 have been calculated over an ensemble of 500 simulation runs, using a different seed value for the random number for each run.

The comparisons of the means (with ±1 SD error bars) of the probability distributions of some key species of the Carletti-Burrage Model using SSA, G-P, the BD-τ method of Chatterjee-Vlachos-Katsoulakis, and the RRA-Noise is shown in Figure 2. The coefficient of variation (CV) for the same species for the same set of simulation methods is shown in Figure 3 below.

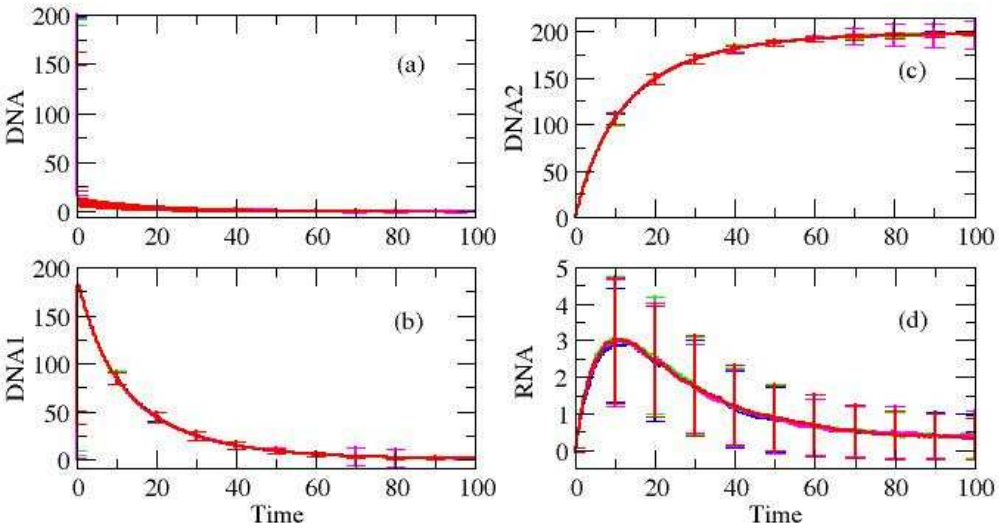

Figure 2. The trajectories of the means with ±1 SD error bars [(a)-(d)] for the probability distributions of the species DNA, DNA1, DNA2 and RNA using SSA (blue curve), G-P (green curve), BD-τ of Chatterjee-Vlachos-Katsoulakis (magenta curve) and RRA-Noise (red curve) for the case of the Carletti-Burrage Model**.**

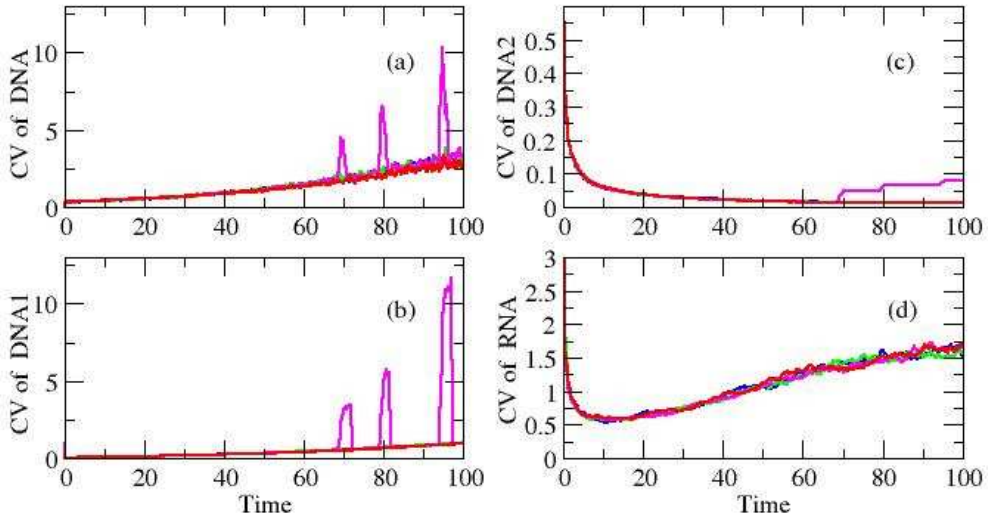

Figure 3. The trajectories of the CVs [(a)-(d)] for the probability distributions of the species DNA, DNA1, DNA2 and RNA using SSA (blue curve), G-P (green curve), BD-τ of Chatterjee-Vlachos-Katsoulakis (magenta curve) and RRA-Noise (red curve) for the case of the Carletti-Burrage Model**.**

The average CPU time and the number of steps taken by the different simulation methods is shown in Table 1.

Table 1. The average values of the CPU times (in seconds) and the number of steps for 500 simulations taken by different simulation methods for the case of Carletti-Burrage Model.

Simulation Methods	SSA	G-P	BD-τ	RRA-Noise
CPU time (sec)	5.029	14.970	40.327	4.234
Steps	33210	30905	16368	2854

The CPU time values in Table 1 show that the newly proposed RRA-Noise is significantly faster than the G-P and the BD-τ methods, and faster than the SSA. This is evident by the less number of steps taken by the RRA-Noise method. The overlap of the trajectories of the means of the respective species in Figure 2 is an indicator of good agreement between the different simulation methods. In the case of the BD-τ of Chatterjee-Vlachos-Katsoulakis (Magenta), it is found that the tail ends of the simulated trajectories (for DNA, DNA1 and DNA2) are not within the ±1 SD error bars of the SSA trajectories. The spikes in profiles of CVs for the same species in Figure 3 are a signature of this deviation, which are not in agreement with the others. The occurrence of the spikes is attributed to the increase in the standard deviation at the respective time points. In case of the BD-τ method, the time steps are taken by employing a coarse grain factor[23], *f*, taken as 2.0. It has been found that the smaller value of the coarse grain factor serves to make the simulations more accurate. However, this also leads to an increase in the CPU time. The increase in the value of "*f*" reduces the CPU time, but this now leads to the loss of accuracy in the simulations. This is shown in Figure S1 of the Supporting Information, where the simulations are performed by increasing the coarse grain factor, *f*, to 4.0. Figure S1 depicts different (inaccurate) simulation trajectories obtained from the BD-τ of Chatterjee-Vlachos-

Katsoulakis at the reduced CPU time. On the other hand, in the case of the G-P method, the choice of SSA during the simulations contributes to the increase in the CPU time. Thus, with the RRA-Noise results lying within the SSA results in terms of ±1 SD error bar, it turns out that it provides good results in terms of accuracy. This is a heartening result, especially since the RRA-Noise is seen to perform considerably better than the BD-τ method, which had been specifically developed to tackle the problem of negative populations in chemical systems.[23]

The Simple Isomerization Reaction Model

The Simple Isomerization Reaction Model consists of the following three reactions.

$$\begin{aligned} R_1 &: X_1 \xrightarrow{c_1} X_2 \\ R_2 &: X_2 \xrightarrow{c_2} X_1 \\ R_3 &: X_2 \xrightarrow{c_2} X_3 \end{aligned} \qquad (2)$$

where X_1, X_2, X_3 are the reacting species and c_1, c_2 and c_3 are the rate constants of the corresponding reactions.

The numerical values of the rate constants of the reactions and initial molecular species are given in Table S10 of the Supporting Information.

Like for the previous example, the Simple Isomerization Reaction Model has also been simulated by different simulation methods. The comparisons of the means with ±1 SD error bars and the CVs of the probability distributions for the species X_1, X_2, X_3 by using the SSA, the G-P, the BD-τ method of Chatterjee-Vlachos-Katsoulakis, and the RRA-Noise is shown in Figure 4. Their trajectories have been calculated over an ensemble of 500 simulation runs.

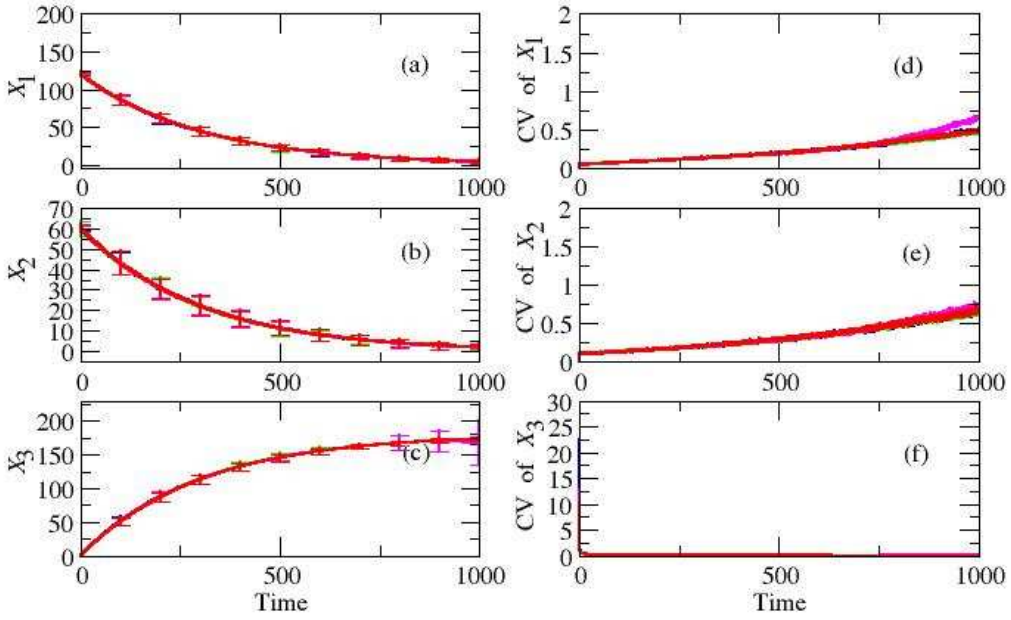

Figure 4. The trajectories of the means with ±1 SD error bars [(a)-(c)] and the CVs [(d)-(f)] for the probability distributions of the species X_1, X_2, X_3 using SSA (blue curve), BD-τ of Chatterjee-Vlachos-Katsoulakis (magenta curve) and RRA-Noise (red curve) for the case of the Simple Isomerization Model**.**

Table 2. The average values of CPU time (in seconds) and the number of steps taken by different simulation methods for the case of the Simple Isomerization Reaction Model.

Simulation Methods	SSA	G-P	BD-τ	RRA-Noise
CPU time (sec)	83.919	33.131	59.479	35.151
Steps	68468	36126	34895	27826

It was found that the simulation of this model by GASA leads to negative numbers, making it inapplicable for comparisons with other methods. The model system has been simulated by the BD-τ method with a coarse grain factor, f, equal to 2.0. It was seen that while the simulated profiles are accurate for this coarse grain factor value, the simulation also takes excess CPU time of 59.479 seconds. An attempt to reduce this CPU time by increasing the value of "*f*" to 50.0 and 100.0 gives totally different trajectories, as shown in Figures S2 and S3 of the Supporting Information. The corresponding values of the CPU times are given in Table S2 and S3. Furthermore, this also comes at the risk of obtaining negative numbers for the species X_2. Thus, for this example, the RRA-Noise again scores over BD-τ: a method that had been developed specifically in order to sort out the issue of negative numbers.

Admittedly, the G-P method is marginally better in terms of accuracy in comparison to RRA-Noise, and the two methods are found to be equally accurate, which indicates that the G-P is the most effective method for the simulation of this particular chemical system. However, the RRA-Noise is only slightly less efficient, which indicates that it would be almost as effective as the G-P in simulating this system. It takes less number of steps than observed for all of the other methods (Table 2). It is also found that the trajectories obtained from the RRA-Noise simulations are within the SSA results from the viewpoint of ±1 SD error bars. Therefore, this example also

showcases the efficiency and reliability of RRA-Noise at simulating a chemical system that is susceptible to the problem of negative numbers.

Simple Model System

The Simple Model system of two reactions discussed here was used by *Cao et al*.[29] to test the reliability and efficiency of their modified Poisson tau leap method. It consists of the following set of reactions:

$$\begin{aligned} R_1 &: X_1 \xrightarrow{c_1} X_2 \\ R_2 &: X_2 \xrightarrow{c_2} X_3 \end{aligned} \qquad (3)$$

where X_1, X_2, X_3 are the reacting species and c_1, c_2 are the rate constants of the corresponding reactions.

The numerical values of the rate constants of the reactions and initial molecular species are given in Table S11 of the Supporting Information.

The study of the associated rate constants of the two reactions ($c_1 = 10, c_2 = 0.1$) and the corresponding reactant species ($X_1 = 9$, $X_2 = 20000$) indicates that there is a possibility of getting negative numbers for the X_1 species. This fact gets confirmed when the G-P method is seen to drive the X_1 species to unrealistic numbers during the simulations. The same is seen to be true for GASA. Hence, apart from the SSA, only BD-τ and RRA-Noise have been considered. The results are shown in Figure 5 and Table 3 below.

In case of species X_1, it was observed that it falls off rapidly and afterwards does not demonstrate any fluctuation. Hence, the time trajectories of the X_1 species are not reported in Figure 5. What is shown are the time trajectories of the X_2 and the X_3 species. For these two species, the values shown in Figure 5 indicate that there is considerable agreement between all the simulation methods.

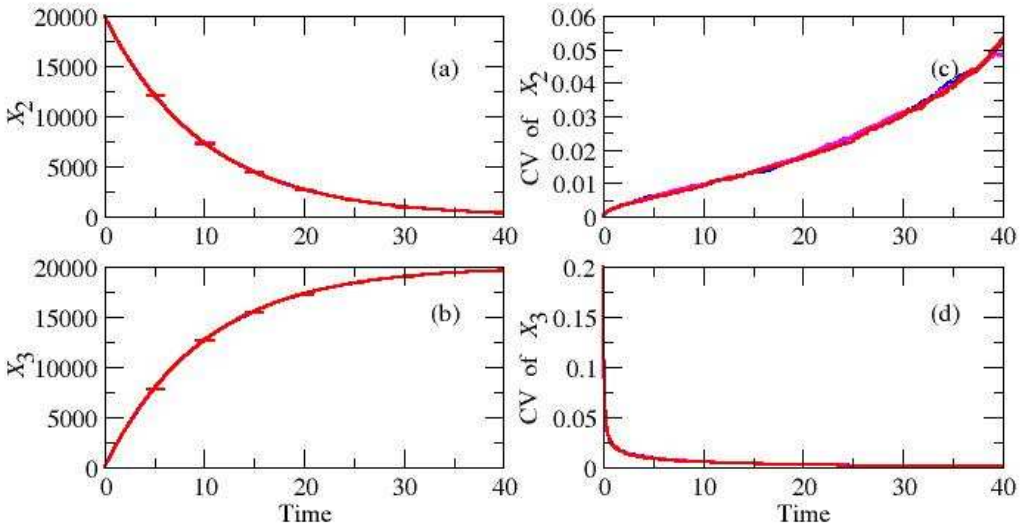

Figure 5. The trajectories of the means with ±1 SD error bars [(a) and (b)] and the CVs [(c) and (d)] for the probability distributions of the species X_2 and X_3 using SSA (blue curve), BD-τ of

Chatterjee-Vlachos-Katsoulakis (magenta curve) and RRA-Noise (red curve) for the case of the Simple Model System**.**

Table 3. The average values of CPU time (in seconds) and the number of steps taken by different simulation methods for the case of the Simple Model System.

Simulation Methods	SSA	BD-τ	RRA-Noise
CPU time (sec)	8.298	16.794	7.257
Steps	19633	9814	726

The mean trajectories of X_2 and X_3 are within the ±1SD error bars of the SSA trajectories. The significantly less number of steps contribute to the CPU time performance of the RRA-Noise. The simulated trajectories and the CPU times tabulated in Table 3 indicate the effectiveness of the RRA-Noise. More importantly, it is again seen to be faster than the BD-τ method. Like earlier examples, any attempt to increase the efficiency of BD-τ leads to loss of accuracy in the results. The results corresponding to different coarse grain values are provided in the Supporting Information.

Model of First Order Reactions: Simulation and Numerical Stability

In this section, the simulation along with its numerical stability of the model of four unimolecular reactions is discussed. This model was used by Chatterjee *et al*.[23] to test their BD-τ method.

$$\begin{aligned} R_1 &: X_1 \xrightarrow{c_1} X_2 \\ R_2 &: X_2 \xrightarrow{c_2} X_3 \\ R_3 &: X_3 \xrightarrow{c_3} X_2 \\ R_4 &: X_2 \xrightarrow{c_4} X_1 \end{aligned} \qquad (4)$$

Here, X_1, X_2 and X_3 are the species taking part in four different reactions and c_1, c_2, c_3 and c_4 are the rate constants of these reactions.

The numerical values of the rate constants of the reactions and the initial molecular species are given in Table S12 of the Supporting Information.

In this model, which consists of all first order reactions, the species X_2 takes part in several of the reactions. This makes the reaction network more complicated in comparison to the earlier example pertaining to the Simple Isomerization Reaction Model.

As shown in Figure 6, there is good agreement between the means and CVs of the probability distributions for the species X_1, X_2 and X_3 obtained by using the SSA, the BD-τ method of Chatterjee-Vlachos-Katsoulakis and RRA-Noise. Unlike the previous two examples, the G-P method gives rise to negative numbers during the simulations as does GASA. Hence, they are not included for the comparative study along with the others. As in all the previous cases, the time profiles of all the species have been calculated over an ensemble of 500 different simulation runs.

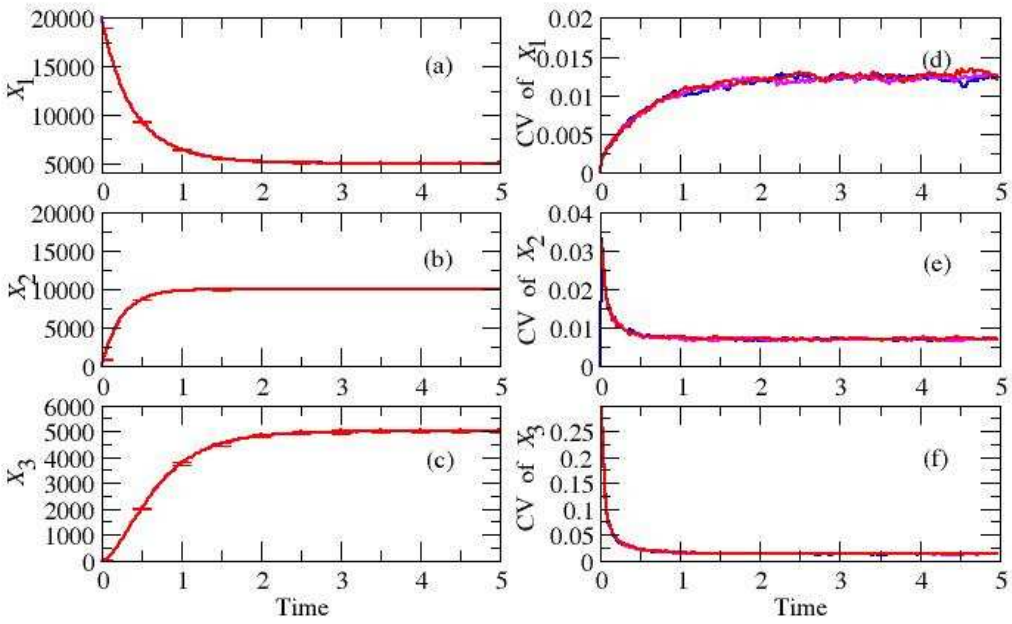

Figure 6. The trajectories of the means with ±1 SD error bars [(a)-(c)] and the CVs [(d)-(f)] for the probability distributions of the species X_1, X_2, X_3 using SSA (blue curve), BD-τ of Chatterjee-Vlachos-Katsoulakis (magenta curve) and RRA-Noise (red curve) for the case of the model consisting of First Order Reactions**.**

Table 4. The average values of CPU time (in seconds) and the number of steps taken by different simulation methods for the case of the model consisting of First Order Reactions.

Simulation Methods	SSA	BD-τ	RRA-Noise

CPU time (sec)	18.141	56.224	9.557
Steps	199180	100001	3640

The average CPU times shown in Table 4 indicates that RRA-Noise is computationally more efficient than the other methods. It is almost twice as fast as the SSA. It also indicates that the number of steps taken by RRA-Noise are very much less than other methods. The low computational efficiency of BD-τ can be attributed to the small size of the time steps, leading to a large number of steps. As before, this can be changed by increasing the value of the coarse grain factor (2.0), but that, as in the previous examples, leads to a significant loss of accuracy. This is illustrated in Figure S4 of the Supporting Information, showing the results of simulations where the coarse grain factor had been increased to 5000.0. As Figure S4 indicates, the mean and the CVs for the different species becomes far less accurate for the BD-τ case in comparison to the other methods. Further increase in the coarse grain factor to 10000.0 also leads to similar results as shown in Figure S5. The CPU values corresponding to the aforementioned f values are given in Table S4 and S5 of the Supporting Information.

The results in Figure 6 provide a good match of simulation methods with each other in terms of ±1 SD error bars. Overall this model of first order reactions is a classic example where, in addition to the accuracy, the CPU times of the corresponding methods are of prime importance. And here, as in the previous cases, the newly proposed RRA-Noise again ends as the most favorable approximate simulation method.

The numerical stability of the RRA-Noise is discussed for this example by taking multiple runs and further benchmarking them against the SSA. The SSA is simulated over an ensemble of 20000 (black curve) simulation runs. The RRA-Noise is simulated over an ensemble of 100 (red curve), 500 (green curve), 1000 (blue curve), 5000 (brown curve) and 10000 (orange curve) simulation runs. It has been found that with the increase in the number of realizations, the RRA-Noise gets converged to the SSA trajectories with the decrease in the error. The error between the trajectories of the SSA and the RRA-Noise has been calculated at some chosen discrete time points. The Figure 7 below shows the trajectories ((a)-(c)) of the species along with their closely monitored behavior ((d)-(f)) on a different scale. The Table 5 provides the absolute errors for the different realizations at specific time points.

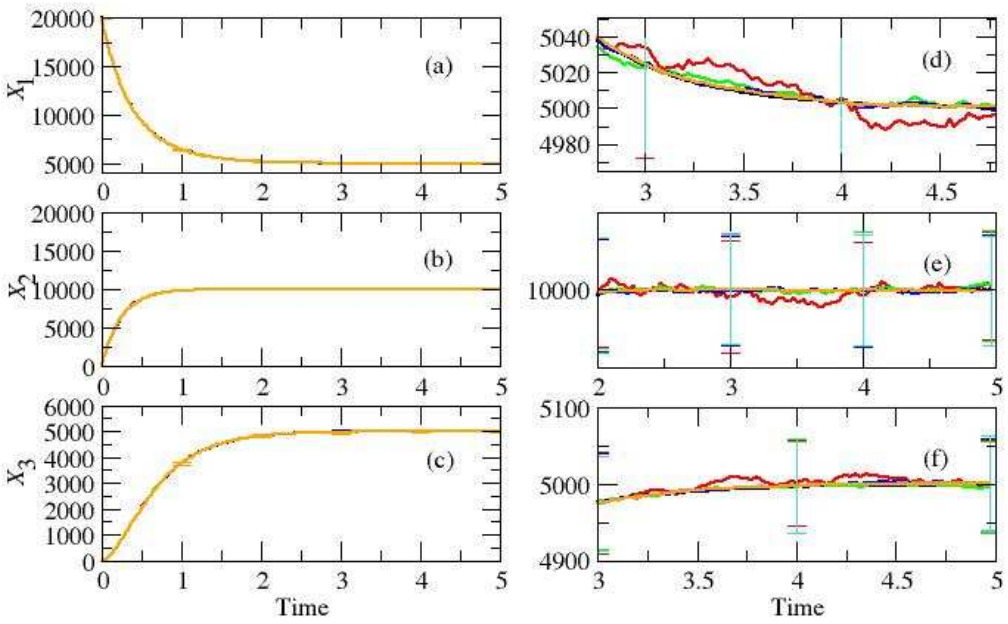

Figure 7. The trajectories of the means with ±1 SD error bars [(a)-(c)] and the same trajectories on a different scale [(d)-(f)] simulated using RRA-Noise with 100 (red curve),500 (green curve),1000 (blue curve),5000 (brown curve),10000 (orange curve) runs and SSA with 20000 runs (black curve) for the case of the model consisting of First Order Reactions**.**

Table 5. The absolute errors between the trajectories of the SSA and the RRA-Noise for different runs at discrete time points.

Time points	100 runs		
1	17.779	2.922	14.856
2	0.294	0.932	0.771
3	9.910	8.100	0.189
4	1.672	5.550	5.878
Time points	500 runs		
1	9.663	5.020	4.642
2	0.577	2.640	4.063
3	1.826	1.895	1.930
4	0.347	2.411	0.063
Time points	1000 runs		
1	8.577	4.830	3.746
2	0.649	1.958	4.608
3	0.672	0.032	1.359
4	0.035	0.580	1.455
Time points	5000 runs		
1	8.379	4.153	4.225
2	1.443	0.923	2.520
3	0.326	2.344	0.672
4	0.212	0.891	1.321
Time points	10000 runs		
1	7.918	3.826	4.092
2	0.728	0.861	1.867
3	0.437	1.230	0.332
4	0.145	0.885	1.260

It is found that with the increase in the number of runs, the absolute error tends to decrease, thereby converging towards the SSA. The trajectory of RRA-Noise with 10000 runs (orange) gets closer to the SSA profile, relative to the curve which has 100 runs (red). This behavior can be observed in the Figures 7 ((d)-(f)), where wide fluctuations are observed for the curve with 100 runs in comparison to those with 10000 runs.

Oscillatory Model System: Simulation and Numerical Stability

The simulation as well as the numerical stability of the oscillatory reaction model, namely the Oregonator model, will be discussed in this section. This model was simulated by Daniel Gillespie by using the SSA. It consists of the following set of reactions:

$$\begin{aligned} R_1 &: \overline{X_1} + Y_2 \xrightarrow{c_1} Y_1 \\ R_2 &: Y_1 + Y_2 \xrightarrow{c_2} Z_1 \\ R_3 &: \overline{X_2} + Y_1 \xrightarrow{c_3} 2Y_1 + Y_3 \\ R_4 &: 2Y_1 \xrightarrow{c_4} Z_2 \\ R_5 &: \overline{X_3} + Y_3 \xrightarrow{c_5} Y_2 \end{aligned} \qquad (5)$$

Here, Y_1, Y_2 and Y_3 are the species taking part in the different reactions, while $\overline{X_1}$, $\overline{X_2}$ and $\overline{X_3}$ signify that the molecular population levels of the species are constant during the simulations. c_1, c_2, c_3, c_4, c_5 are the rate constants of the reactions.

The numerical values of the rate constants of the reactions and initial molecular species are given in Table S13 of the Supporting Information file. The simulation of this chemical system by GASA leads to negative numbers, hence the results with the GASA have not been discussed further. The BD-τ of Chatterjee-Vlachos-Katsoulakis has been found to be inapplicable for this particular system. The rest of the methods: SSA, G-P and RRA-Noise have been discussed below. The oscillatory nature of this model poses a challenge to methods that claim to solve the problem of negative numbers.

The mean trajectories are shown in Figures 8(a)-8(c), while Figures 8(d)-8(f) show the corresponding trajectories of the CVs. The behavior of the trajectories in the Figures 8(a)-8(c) by the different simulation methods underlines the oscillatory nature of the chemical system. The

trajectories of the G-P (green curve) show a slightly out-of-phase behavior relative to the others. However, the trajectories of SSA (blue curve) and RRA-Noise (red curve) are found to be in good agreement. This is observed to a good extent in all the curves.

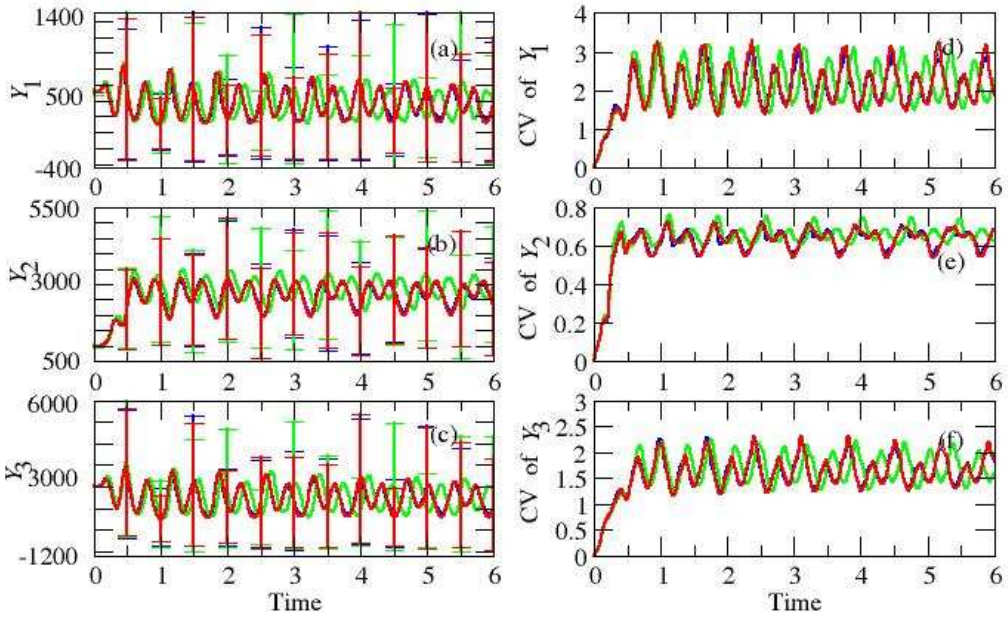

Figure 8. The trajectories of the means with ±1 SD error bars [(a)-(c)] and of the CVs [(d)-(f)] for the probability distributions of the species Y_1, Y_2, Y_3 using SSA (blue curve), G-P (green curve) and RRA-Noise (red curve).

Table 6. The average values of CPU time (in seconds) and the number of steps taken by different simulation methods for the case of the Oregonator Model.

Simulation Methods	SSA	G-P	RRA-Noise
CPU time (sec)	70.069	14.976	35.224
Steps	694868	16542	29125

The comparison of the CPU times of different methods in Table 6 indicates that the G-P is more efficient than the rest of the methods. However, their simulated profiles indicates a slight outward shift relative to the trajectories of other methods. It is also observed that the G-P trajectories are not within the SSA trajectories in terms of ±1 SD error bars. On the other hand, the RRA-Noise is found to take relatively more steps, but is able to reproduce the trajectories accurately. More importantly, no negative molecular numbers have been found during the simulations.

The numerical stability of this model system is discussed along the lines similar to those of the previous example. The given model is simulated by the SSA using an ensemble of 20,000 (black curve) simulation runs, while the RRA-Noise is used for the simulation over an ensemble of 100 (red curve), 500 (green curve), 1000 (blue curve), 5000 (brown curve) and 10,000 (orange curve) simulation runs. Unlike the previous example, this is an oscillatory model with no steady state. It is found that the absolute error at some discrete points decreases with an increase in the number of realizations of the RRA-Noise.

It has been found that with the increase in the number of realizations, the RRA-Noise gets converged to the SSA trajectories with decrease in the error. The error between the trajectories of the SSA and the RRA-Noise has been calculated at some chosen discrete time points. Figure 9 below shows the mean trajectories ((a)-(c)) of the species, while Table 7 provides the absolute errors for the different realizations at specific time points.

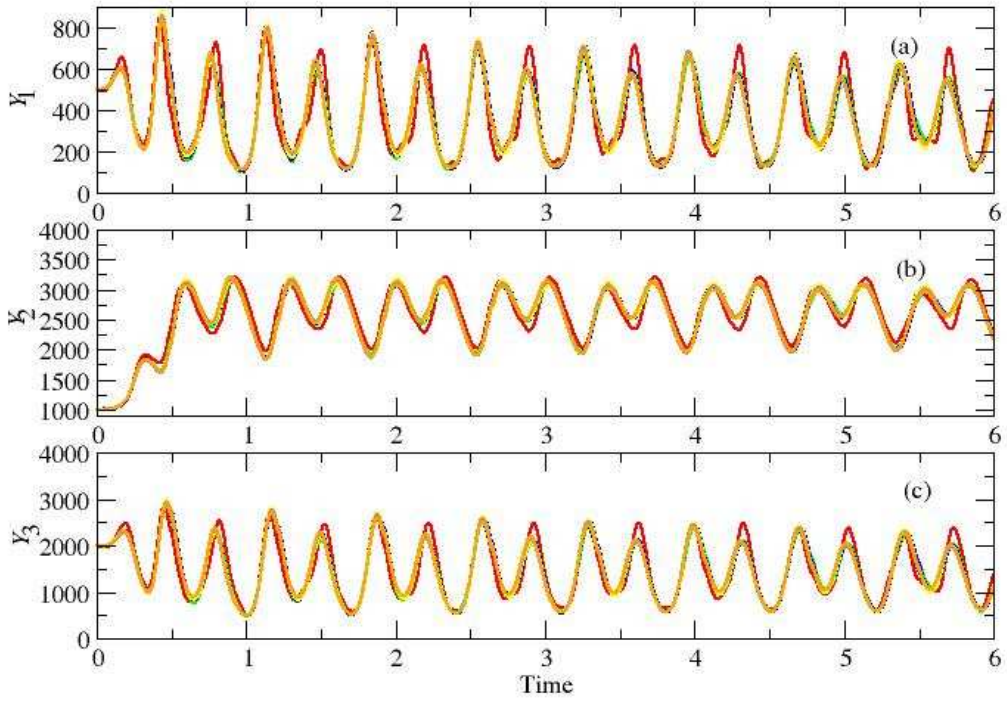

Figure 9. The trajectories of the means [(a)-(c)] simulated using the RRA-Noise with 100 (red curve), 500 (green curve), 1000 (blue curve), 5000 (brown curve), 10,000 (orange curve) runs and the SSA with 20,000 runs (black curve), for the case of the Oregonator Model.

Table 7. The absolute errors between the trajectories of the SSA and the RRA-Noise for different runs at discrete time points.

Time points	100 runs		
1	11.396	161.357	18.711
2	22.308	48.094	194.045
3	2.155	11.085	160.426
4	0.985	52.447	43.190
Time points	500 runs		
1	3.174	6.695	17.408
2	41.302	111.648	77.653
3	11.188	72.191	49.520
4	49.485	41.032	29.818
Time points	1000 runs		
1	2.001	21.669	8.416
2	30.584	109.157	22.778
3	7.388	56.546	34.213
4	49.250	38.105	94.080
Time points	5000 runs		
1	0.979	47.324	2.136
2	6.969	61.696	5.923
3	17.139	9.866	61.658
4	22.457	24.735	84.764
Time points	10000 runs		
1	1.304	14.978	8.993
2	3.192	9.973	0.945
3	13.437	26.698	47.993
4	5.367	25.048	10.685

In Figure 9 above, it is observed that the simulation by the RRA-Noise with 100 runs (red curve) shows a downward and upward shift relative to the SSA trajectory with 20,000 runs (black curve). The trajectory with 10,000 runs (orange curve) gets substantially closer to the exact SSA trajectory. The variation in the absolute error in Table 7 is seen as a signature of the highly oscillatory character of the chemical system.

Overall, from the simulations of the five different examples, discussed in the sections above, it can be speculated that the newly developed method, which uses the RRA and accounts for noise during the simulation solves the problem of negative populations. In all the examples considered, it was seen that, unlike in the RRA-Noise case, one has to find the best possible value of the coarse grain factor for the BD-τ method of Chatterjee *et al.*[21] in order to achieve the necessary accuracy, a choice that usually led to a loss of efficiency for the method.

The examples that have been chosen and discussed in the current work were those that highlight difficult cases where the existing state-of-the-art methods either fail or perform with lower efficiency. However, it is to be noted that the current approach is not a general theoretical modification in stochastic simulations for correcting the number of firings for every leap during the simulations, but is a remedy to the negative population problem for specific leaps where the reactant population becomes negative due to the wrongly calculated number of firings, for those leaps, by existing methods. Therefore, it is still possible that the current method, while clearly having been demonstrated to have performed well for the examples considered, might also provide negative numbers for reactant population in certain cases, and thus fail for certain chemical systems. It is, nevertheless, expected, that the current recipe for correcting the problem of negative populations would work in a large majority of cases, as the current set of examples demonstrates.

Conclusions

In order to achieve a speed-up over the SSA, various approximate accelerated methods have been developed. However, such approaches are fraught with problems of accuracy,

problems that become more acute when dealing with chemical systems that deal with low molecular populations. In such cases, there are instances where negative molecular numbers have been obtained during the simulations. In the current work, we have sought to solve this problem by introducing the novel concept of accounting for the "noise" obtained for the number of firings of each reaction in a given time step. We have tested out this idea by combining the "noise accounting" with the accelerated method: the Representative Reaction Approach (RRA) that we had developed earlier[21]. This new method, termed as "RRA-Noise", has been tested on a number of different examples, ranging from simple unimolecular system to oscillatory chemical system. It has been found that the RRA-Noise is effective not only in terms of accuracy but also in efficiency, in comparison to state-of-the-art approximate accelerated methods such as Gillespie's Approximate Stochastic Algorithm (GASA)[15], Gillespie-Petzold (G-P)[16], BD-τ of Chatterjee *et al*.[23] This newly developed method has the added virtue of being quite simple and easy to code. The discussion pertaining to the stability of algorithm emphasizes the robustness of newly proposed method. Furthermore, for the newly proposed method, there is no necessity to change the value of error control parameter for every new chemical system that has to be simulated. Finally, it may also be mentioned that the notion of accounting for noise during a simulation may find applications in other fields of interest as well.

Supplementary Information Available: The Fortran 95 codes for the different algorithms discussed in the text, including the SSA, G-P, BD-τ of Chatterjee-Vlachos-Katsoulakis, RRA-Noise are provided.

Acknowledgements

The authors acknowledge financial support from Department of Science and Technology (DST), India. The authors also acknowledge the Multi-Scale Simulation and Modeling project - MSM - for providing financial assistance. SK thanks CSIR for SRF.

References

[1] D. A. McQuarrie, *J. Appl. Probab.* **1967,** *4*, 413.
[2] D. T. Gillespie, *Physica A* **1992**, *188*, 404.
[3] A. B. Bortz, M. H. Kalos, and J. L. Lebowitz, *J. Comput. Phys.* **1975**, *17*, 10.
[4] D.T. Gillespie, *J. Comput. Phys.* **1976**, *22*, 403.
[5] D. T. Gillespie, *J. Phys. Chem.* **1977**, *81*, 2340.
[6] M. A. Gibson, J. Bruck, *J. Phys. Chem. A* **2000**, *104*, 1876.
[7] Y. Cao, H. Li, L. Petzold, *J. Chem. Phys.* **2004**, *121*, 4059.

[8] J. M. McCollum, G. D. Peterson, C. D. Cox, M. L. Simpson, N. F. Samatova, *Comput. Biol. Chem.* **2006**, *30*, 39.
[9] C. Yates, G. Klingbeil, *J. Chem. Phys.* **2013**, *138*, 094103.
[10] M. Barrio, K. Burrage, A. Leier, T. Tian, *PLoS Comput. Biol.* **2006**, *2*, 117.
[11] E. L. Haseltine, J. B. Rawlings, *J. Chem. Phys.* **2002**, *117*, 6959.
[12] A. Hellander, P. Lötstedt, *J. Comput. Phys.* **2007**, *227*, 100.
[13] H. Salis, Y. Kaznessis, *J. Chem. Phys.* **2005**, *122*, 054103.
[14] D. T. Gillespie, *J. Chem. Phys.* **2000**, *113*, 297.
[15] D. T. Gillespie, *J. Chem. Phys.* **2001**, *115*, 1716.
[16] D. T. Gillespie, L. R. Petzold, *J. Chem. Phys.* **2003**, *119*, 8229.
[17] M. Rathinam, L. R. Petzold, Y. Cao, D. T. Gillespie, *J. Chem. Phys.* **2003**, *119*, 12784.
[18] Y. Cao, D. T. Gillespie, L. R. Petzold, *J. Chem. Phys.* **2006**, *124*, 044109.
[19] Y. Xu, Y. Lan, *J. Chem. Phys.* **2012**, *137*, 204103.
[20] X. Cai, Z. Xu, *J. Chem. Phys.* **2007**, *126*, 074102.
[21] S. Kadam, K. Vanka, *J. Comput. Chem.* **2012**, *33*, 276.
[22] T. Tian, K. Burrage, *J. Chem. Phys.* **2004**, *121*, 10356.
[23] A. Chatterjee, D. G. Vlachos, M. A. Katsoulakis, *J. Chem. Phys.* **2005**, *122*, 024112.
[24] M. F. Pettigrew, H. Resat, *J. Chem. Phys.* **2007**, *126*, 084101.
[25] X. Peng, W. Zhou, Y. Wang, *J. Chem. Phys.* **2007**, *126*, 224109.
[26] A. Auger, P. Chatelain, P. Koumoutsakos, *J. Chem. Phys.* **2006**, *125*, 084103.

[27] A. Leier, T. T. Marquez-Lago, K. Burrage, *J. Chem. Phys.* **2008**, *128*, 205107.

[28] S. Kadam, K. Vanka, *J. Comput. Chem.* **2013**, *34*, 394.
[29] Y. Cao, D. T. Gillespie, L. R. Petzold, *J. Chem. Phys.* **2005**, *123*, 054104.
[30] C. Yates, K. Burrage, *J. Chem. Phys.* **2011**, *134*, 084109.
[31] C. W. J. Beenakker, M. Buttiker, *Phys. Rev. B* **1992**, *46*, 1889.
[32] Ya. M. Blanter, M. Buttiker, *Phys. Rep.* **2000,** *336*, 1.

[33] W. Press, B. Flannery, S. Teukolsky, W. Vetterling, Numerical Recipes: The Art of Scientific Computing; Cambridge University Press: Cambridge, **1986**.

An attempt to reduce the computational time of the stochastic simulations of chemical kinetics leads to negative numbers, which in turn gives inaccurate simulation trajectories. A computational method based on the concept of noise in conjunction with the representative reaction approach is proposed to solve this problem. It has been found that, the new method performs better on the front of accuracy and efficiency than other state-of-the-art methods.