Hybrid, Multiscale Monte Carlo Algorithm for Simulating Stochastic Systems

Asawari Samant, Babatunde Ogunnaike and Dionisios Vlachos
Department of Chemical Engineering, University of Delaware

INTRODUCTION

Accurately relating the physiological outcome of a cell to the molecular events, based on an *in silico* analysis of intracellular networks, requires not only a precise knowledge of the network, but also an appropriate simulation technique. Traditionally, the methods used to temporally evolve intracellular networks have been deterministic in nature. However, deterministic methods cannot account for the microscopic randomness stemming from the low population of some species in intracellular environments. Hence, in situations where intracellular noise is critical to the functional behavior of the cell, deterministic solvers fare poorly. Exact stochastic solvers like the stochastic simulation algorithm (SSA)[1] which evolve systems by accounting for each microscopic event occurring in the system, are an excellent alternative to simulate such noisy systems.

The systematic recording of microscopic events, which is responsible for the exactness of stochastic algorithms, is also the main cause for its inefficiency. Evolving the system one reaction at a time is difficult, especially when large networks and/or large populations are encountered. τ-leap methods,[2-4] which perform multiple firings of the reactions in a pre-decided time interval, overcome this drawback of the exact SSA. However, the τ-leap methods[2-4] cannot handle low population, since both the efficiency and accuracy of the τ-leap methods scale down with population. The hybrid τ-leap-SSA[5] method tackles this challenge by switching between the exact SSA at small population and the coarse-grained τ-leap method at large population. Using this adaptive technique, the hybrid algorithm[5] can handle mixed population levels in the reaction network.

Exact stochastic algorithms,[6] as well as the coarse-grained τ-leap methods[2-4] , are frustrated by the separation of time scales that are commonly encountered in the intracellular networks. The challenge posed by separation of time scales in a stochastic framework is identical to the problem of numerical stiffness faced by deterministic solvers. The reaction sampling method in stochastic algorithms results in the frequent execution of the fast reactions, without a significant increment in time. Several stochastic multiscale algorithms[7-14] introduced to handle disparity of time scales in a stochastic framework, try to reduce the time spent by the algorithms in simulating the fast network. Some of the multiscale algorithms make *à priori* assumptions about the populations in the fast networks, and use an approximate stochastic method like the Langevin method to accelerate the simulation of the fast network[10]. This approach fails when the faster scales in the network arise purely from the large magnitude of the reaction rate constants.

A more generic multiscale approach, applies the quasi-steady state or quasi-equilibrium (QE) approximation, which is popular in the deterministic treatment of stiff networks, to the chemical master equation (CME).[7,9,12,14] Similar to QSSA in deterministic solvers, this approach entails partitioning the network, obtaining a probabilistic description of the quasi-equilibrium of the fast network, and using this quasi-equilibrium description to evolve the slower scales in the network. While conceptually simple, the probabilistic nature of a stochastic framework, and the discrete nature of reacting chemical species, makes the implementation of this stochastic multiscale approach challenging. For example, quasi-equilibrium in a stochastic framework is characterized by a probability distribution function, which is difficult to obtain. The analytical framework, suggested by Cao et al.[7,8] restrict the applicability of the approach to very simple systems. On the other hand, a mean-field approximation[7,8] of the quasi-equilibrium will propagate errors in the evolution of the slow network, when the QE PDF involves low population or displays multimodality. To overcome these drawbacks, Samant and Vlachos[9] introduced the multiscale Monte Carlo algorithm that numerically generates the PDF of the relaxed fast scales in a numerical manner. The MSMC[9] uses the exact SSA (microscopic solver) on the fast network, to sample the relaxed state space, and obtain a numerical approximation of the QE PDF. This numerically generated QE description is used in the macroscopic solver to evolve the slow network. However, since the individual time scales are still described by the exact SSA, the presence of large populations in either or both the partitioned networks cannot be handled effectively by this multiscale strategy.

A generic stochastic algorithm, which simultaneously addresses the disparity in time scales and population levels, is currently lacking. In this talk, we propose a hybrid, multiscale Monte Carlo algorithm to fill this void. The novelty of this work lies in its ability to simultaneously handle separation of scales in two dimensions, namely the time scales and the populations. More importantly, similar to the original multiscale Monte Carlo algorithm,[9] the proposed algorithm does not make any *a priori* assumption about the scales in the simulated network. The absence of such an assumption, allows the algorithm to effortlessly collapse to the exact and/or non-multiscale stochastic solver, when a separation of scales is absent. To the best of our knowledge, such a generic and seamless treatment of separation of time scales and population is currently absent. Additionally, we have introduced a new relaxation criterion that eliminates the need for the complete description of the quasi-equilibrium, further reducing the computational expense of the method. Obtaining the complete quasi-equilibrium (QE) probability distribution function is a computationally intensive task, especially when the accessed quasi-equilibrium state space is vast. Combining the new relaxation criterion with the hybrid solvers results in an unprecedented speedup.

We begin the talk with a brief introduction to the original multiscale Monte Carlo method,[9] which used the exact SSA as the microscopic and macroscopic solvers, and highlight some of its limitations. Next, we introduce the hybrid multiscale Monte Carlo method, that overcomes these limitations of the MSMC method at high population levels. Finally, using a few biological examples, we demonstrate the efficiency and accuracy of the HyMSMC method.

SSA notation :

Consider a well-mixed, isothermal system of n species belonging to species set $\boldsymbol{S} \equiv \{S_1, S_2, ..., S_n\}$ reacting through m reaction from reaction set $\boldsymbol{R} \equiv \{R_1, R_2, ..., R_m\}$. Let the state of the system at any time t be denoted by a n-dimensional vector $\mathbf{X}(t) = (X_1(t), X_2(t), ..., X_n(t))$, where $X_i(t)$ is the number of molecules of species S_i at time t. n-dimensional vector $\boldsymbol{\upsilon}_j$ corresponds to the stoichiometry vector of reaction R_j, such that υ_{ij} is the stoichiometric coefficient of species S_i in reaction R_j. Given that the system is in a state $\mathbf{X}(t) = \mathbf{x}$ at time t, we define a propensity function

$a_j(\mathbf{x},t)$ such that $a_j(\mathbf{x},t)dt$ gives the transition probability of the j^{th} reaction, R_j, occurring in an infinitesimally small time interval (t, t+dt). This function is typically dependent on the state of the species, and the reactions conditions of the network such as temperature, pressure.

THE MULTISCALE MONTE CARLO METHOD

The multiscale Monte Carlo (MSMC) [9]method is primarily made of four steps that are implemented in an iterative manner.

1. Partition the reactions and species based on an user-defined cut-off on the reaction propensities
2. Using an exact stochastic solver such as the SSA, evolve and relax the fast network to its quasi-equilibrium distribution.
3. Generate the numerical approximation of the quasi-equilibrium PDF by running the stochastic solver on the relaxed fast network.
4. Use the quasi-equilibrium description of the fast network in an appropriate manner, to evolve the slow network by a macroscopic time step.

Steps 2 and 3 constitute the microscopic solver and step 4 constitutes to the macroscopic solver. In the following section, we brief explain the four key aspects in the multiscale Monte Carlo method.

Partitioning the reaction network

The separation of time scales in a well-mixed reaction environment implies the presence of a few reactions in the network that are much faster than the rest. The first step in partitioning a reaction network on the basis of time scales, involves identifying these fast and slow reactions subset, $\boldsymbol{R}^f$ and $\boldsymbol{R}^s$, respectively. Having partitioned the reaction network, we next classify the species as either fast or slow species subsets, $\boldsymbol{S}^f$ and $\boldsymbol{S}^s$, respectively. This partitioning scheme was first proposed by Cao et al[7]. All the species participating in the fast reactions are defined as fast species, and the rest are defined as slow species. Notice the asymmetry in the classification of the species; while the fast species may participate in the slow reactions, the slow species are absent in the fast network. The participation of the fast species in the slow reactions couples the slow and the fast network, thus making the multiscale treatment of the stiff network challenging.

Stochastic quasi-equilibrium criterion

For the partitioned fast network to be conducive to a stochastic quasi-equilibrium treatment, it must have a stable, physically feasible equilibrium at any state, $\mathbf{x}(t)$. [7]In a stochastic framework, the criterion for the stable equilibrium of the isolated fast network is given by [7]

$$\lim_{t'\to\infty} \hat{P}(\mathbf{x^f}, t' \mid \mathbf{x}, t) = \hat{P}(\mathbf{x^f}, \infty \mid \mathbf{x}) \qquad (1)$$

$\hat{P}(\mathbf{x^f}, t')$ is the probability of observing $\hat{\mathbf{X}}^f(t') = \mathbf{x^f}$, where $\hat{\mathbf{X}}^f(t)$ is a stochastic process that is identical to $\mathbf{X}^f(t)$, but evolves the fast species purely via the fast network. Since the slow processes cannot be switched off in reality, the stochastic process, $\hat{\mathbf{X}}^f(t)$, was termed the *virtual fast process* by Cao et al. [7] $\hat{P}(\mathbf{x}^{\mathbf{f}},\infty\,|\,\mathbf{x})$ is the stationary or time-invariant distribution characterizing the relaxed fast network at state **x** of the network.

Eq. (1) merely implies that the fast network is capable of relaxing to a time-invariant distribution given by $\hat{P}(\mathbf{x}^{\mathbf{f}},t')$. Employing this quasi-equilibrium description of fast network to evolve the slow network will be computationally advantageous only if it can be obtained with a relatively small computational cost. Mathematically, this requirement is given by

$$\tau^{f}_{Rel} \ll \min_{j\in \boldsymbol{R}^{s}}\left(\tau^{s}_{j}\right) \qquad (2)$$

where, τ^{f}_{Rel} is the relaxation time of the fast network, and $\min_{j\in \boldsymbol{R}^{s}}\left(\tau^{s}_{j}\right)$ is the smallest time scale in the slow network. Eq. (2) imposes a requirement on the minimum separation of time scales present in the reaction network. Provided Eq. (2) is satisfied, Eq. (1) can be rewritten as,

$$\lim_{t'\to\tau^{f}_{Rel}} \hat{P}(\mathbf{x}^{\mathbf{f}},t'\,|\,\mathbf{x},t\;) \approx \hat{P}(\mathbf{x}^{\mathbf{f}},\infty\,|\,\mathbf{x})$$

SSA as the microsolver

In the last section, we established the two prerequisites for extending the quasi-equilibrium approximation to a stochastic system. Next, we need a way to obtain this probabilistic description of the relaxed fast network. For very simple fast networks, the quasi-equilibrium PDF can be obtained analytically. But this approach has limited applicability. Alternatively, Cao et al. [7]suggested a mean-field approximation of the relaxed low dimensional manifold (LDM) that neglects the stochasticity of the relaxed fast network. This approach fails when the noise on the LDM is critical to the correct evolution of the slow network. Thus, the correct implementation of the stochastic quasi-equilibrium approximation is dependent on a good interfacing strategy between the fast and slow scales.

In the multiscale Monte Carlo method[9], we proposed using a stochastic time integrator to obtain the distribution of the relaxed fast network. We used the SSA as the stochastic solver to relax the fast network at any state **x**(t) and obtain the quasi-equilibrium description. Since the time steps in the evolution of the fast network are relatively small, the solver used to evolve the fast network is called the *microscopic solver*. The stationary PDF, $\hat{P}(\mathbf{x}^{\mathbf{f}},\infty\,|\,\mathbf{x})$, describing the relaxed fast network can be numerically obtained as follows

$$\hat{P}\left(\mathbf{x}^{\mathbf{f}},\infty\,|\,\mathbf{x}\right) \approx \frac{\Delta t^{f}}{\tau_{Eqm}} \qquad (3)$$

where, Δt^{f} is the total life time of the state $\mathbf{x}^{\mathbf{f}}$ in an equilibrium simulation of the fast network for time $\tau_{Eqm} = \sum_{x^{f'}} \Delta t^{f'}$, given that the fast network is relaxed from a state **x**. Δt^{f} is obtained by using the microscopic solver to relax and evolve the fast network. Note that we are evaluating the probabilities based on the life-time of the states in the relaxation period, τ_{Eqm}, instead of their frequencies. Using a PDF evaluated based on the frequency of the states visited during the equilibrium period results in an erroneous evolution of the slow network.

SSA as the Macrosolver

Relaxing the fast network, and obtaining the quasi-equilibrium description is the first step in a stochastic multiscale treatment. Next, the MSMC method uses this probabilistic description of the relaxed fast network to evolve the slow network. Since the fast species may participate in the slow reaction as reactants, the distribution of states of the fast species in the quasi-equilibrium regime translates into a distribution of the propensities of the slow reactions. One way to incorporate the complete probabilistic distribution of the quasi-equilibrium state space, is evaluating a mean of the propensities of the slow reaction over the stationary PDF.

$$\overline{a_j^s}(\mathbf{x}) = \sum_{x^{f'}} \hat{P}(\mathbf{x^{f'}}, \infty \mid \mathbf{x}) \cdot a_j^s\left(\mathbf{x^{f'}}, \mathbf{x^s}\right) \tag{4}$$

$\overline{a_j^s}(\mathbf{x})$, called the slow-scale propensity, is the average of the propensity, $a_j^s\left(\mathbf{x^{f'}}, \mathbf{x^s}\right)$, evaluated over the stationary PDF, $\hat{P}(\mathbf{x^{f'}}, \infty \mid \mathbf{x})$. The *slow-scale approximation* proposed by Cao et al. [7]uses the slow-scale propensities as effective transition probabilities to evolve the slow network after relaxing the fast network.

Using (3) in (4), we get

$$\overline{a_j^s}(\mathbf{x}) = \frac{\sum_{x^{f'}} \Delta t^{f'} \cdot a_j^s\left(\mathbf{x^{f'}}, \mathbf{x^s}\right)}{\tau_{Eqm}}, \text{ for } j = 1, 2, ..., m^s \tag{5}$$

The question that naturally follows from the (5), is how does one gauge the relaxation of the network, and estimate in advance the time, τ_{Eqm}, for which the relaxed fast network needs to be simulated, to get an accurate estimate for the slow-scale propensities.

Relaxation criterion

In the original MSMC algorithm, we generated the complete probabilistic description of the quasi-equilibrium state space at every macroscopic time step. This is computationally expensive especially when the quasi-equilibrium state space is vast. As we saw in the previous section, the only information that is passed from the microscopic solver to the macroscopic solver is the value of the slow-scale propensities. Thus, provided we have sufficiently converged the slow-scale propensities, we should be able to correctly evolve the slow network. Using this observation, we use the convergence of the slow-scale propensities as an approximate criterion for relaxation of the fast network. In doing so, we are ensuring the convergence of only the first moment of the distribution of the fast states. We use the simple 2 sample t-test[15] to test the convergence of the slow scale propensities.

A 2 sample t-test is a commonly used statistical tool to verify with a certain confidence, if two independently drawn samples of a random variable belong to the same probability distribution. In the current context, the state of the fast species (and also the slow reaction propensity a_j^s) is the random variable, such that a sample of a size 'p' corresponds to 'p' events of the fast network, with the state at every fast event corresponding to one data point in the sample. The t-statistic for a 2 sample t-test[15], assuming unequal sample size N_1 and N_2, and unequal variances is given by

$$\boldsymbol{t_j} = \frac{\left.\overline{a_j^s}\right|_1 - \left.\overline{a_j^s}\right|_2}{\sqrt{\frac{\left.\sigma_j^2\right|_1}{N_1} + \frac{\left.\sigma_j^2\right|_2}{N_2}}}, \quad \forall\ j \in \boldsymbol{R^s} \tag{6}$$

$\left.\overline{a_j^s}\right|_1$ and $\left.\sigma_j^2\right|_1$ are expectation and variance of the propensity a_j^s evaluated from $N_1 = (w-1)\cdot MCE_{Win}$ MC events in the first (w-1) simulation windows. $\left.\overline{a_j^s}\right|_2$ and $\left.\sigma_j^2\right|_2$ are expectation and variance of the propensity a_j^s from $N_2 = MCE_{Win}$ events in the w^{th} (last) window. MCE_{Win} is the number of MC events in each window. The slow-scale propensities are said to be converged to a stationary value when

$$-\boldsymbol{t}_{\alpha/2,\upsilon} < \boldsymbol{t_j} < \boldsymbol{t}_{\alpha/2,\upsilon}, \quad \text{for all } j \in \boldsymbol{R^s} \tag{7}$$

where, $\boldsymbol{t}_{\alpha/2,\upsilon}$ is the value of the t-statistic for υ degrees of freedom and a significance level of $\alpha/2$. υ is evaluated as

$$\upsilon = \frac{\left(\left.\sigma_j^2\right|_1 \big/ N_1 + \left.\sigma_j^2\right|_2 \big/ N_2\right)^2}{\left(\left.\sigma_j^2\right|_1 \big/ N_1\right)^2 \big/ (N_1 - 1) + \left(\left.\sigma_j^2\right|_2 \big/ N_2\right)^2 \big/ (N_2 - 1)} \tag{8}$$

To simplify the implementation of the above test and to avoid the recurring evaluation of the $t_{\alpha/2,\upsilon}$, we assume that $\upsilon > 40$ and significance level α=0.05, and hence fix the critical value at $\boldsymbol{t}_{\alpha/2,\upsilon} \approx 1.96$. Since $\boldsymbol{t}_{\alpha/2,\upsilon}$ increases with decreasing υ, using $\boldsymbol{t}_{\alpha/2,\upsilon} \approx 1.96$ in cases where $\upsilon < 40$ imposes a more stringent criterion on the relaxation, and hence can be safely used.

Limitations of the MSMC method

Since the MSMC method uses the exact SSA to evolve both the partitioned networks, it is inefficient at large populations due to the inherent inefficiency of the SSA in handling these conditions. Relaxing the fast network to its quasi-equilibrium using the SSA becomes expensive when the network size is large. On the other hand, when the populations involved in the slow network are large, a single slow event may not perturb the fast network too far away from its low-dimensional manifold. Consequently, it is not necessary to recalculate the slow-scale propensities at every macroscopic state. As a result, it might be possible to reuse the same slow-scale propensities for a few macroscopic events without sacrificing the accuracy of the multiscale approach.

Using a coarse-grained solver for the fast and slow network can efficiently handle both these issues arising from large populations. However, a coarse-grain solver, like the τ–leap method cannot handle small population. So, we propose the hybrid τ–leap-SSA solver[5] as the microscopic, as well as, the macroscopic solver to deal with the mixed populations in either or both the partitioned networks. We call the new multiscale scheme using the hybrid solvers, the *hybrid multiscale Monte Carlo* algorithm.

THE HYBRID MULTISCALE MONTE CARLO ALGORITHM

The hybrid multiscale Monte Carlo differs from the original MC algorithm in primarily one way – the use of the hybrid solver instead of the exact SSA as microscopic and macroscopic solvers. The hybrid solver was introduced by Cao et al. [5] as a modification to the τ–leap methods, to avoid negative population predicted by the τ–leap methods at low population. Since the hybrid solver reduces to the SSA when all the populations are low, the HyMSMC is capable of collapsing to the MSMC in an adaptive manner.

Hybrid solver as the microscopic solver

The use of the hybrid stochastic solver as the microscopic solver is quite straightforward. Using the hybrid solver as the microsolver mainly accelerates the relaxation of the fast network via a coarser sampling of the quasi-equilibrium state space of the fast species. Incorporation of the hybrid solver as the microsolver, begins with classification of the fast reactions into a SSA reactions and TL reactions subset, $\boldsymbol{R}_{SSA}^{f}$ and $\boldsymbol{R}_{TL}^{f}$ respectively. All the fast reactions whose reactant(s) population is less than some critical population X_{crit}, are defined as SSA reactions. The intention of such a classification is that no more than one reaction from the SSA reaction group is executed, thus eliminating the chance of seeing negative populations. The average wait time till the occurrence of the next fast reaction belonging to $\boldsymbol{R}_{f}^{SSA}$ is evaluated as

$$\tau_{SSA}^{f} = -\frac{\ln(\xi_1)}{\sum\limits_{j \in \boldsymbol{R}_{SSA}^{f}} a_j^f} \tag{9}$$

We suggest X_{crit} in the population range 5-20, and have successfully used $X_{crit}=10$ in most of simulations shown in the results section. The cost and accuracy of the hybrid solver increases with X_{crit}, with the method reducing to SSA for $X_{crit} \to \infty$, and to a Poisson τ–leap[2] for $X_{crit}=0$.

The hybrid solver executes multiple firings of the TL reactions in the leap interval. The leap time is estimated from the leap criterion that imposes an upper bound on the expectation and the variance of the change in the propensity in this leap time. In this paper we use the simpler, r-criterion introduced by Chatterjee et al. [3] to evaluate τ_{TL}^{f}. The criterion given by

$$\tau_{TL}^{f} = \min_{i \in \boldsymbol{S}^f} \left\{ \frac{r x_i^f}{\sum\limits_{j \in \boldsymbol{R}_{TL}^{f};\, \upsilon_{ij}^f < 0} -\upsilon_{ij}^f a_j^f\left(\mathbf{x}^f\right)} \right\}, \tag{10}$$

places an upper bound on the expected consumption of species through the leaping reactions. Here, the user-defined parameter r that decides the maximum limit on the fractional change. The leap time $\boldsymbol{\tau}^f$ is chosen as

$$\tau^f = \min\left\{\tau_{TL}^f, \tau_{SSA}^f\right\} \tag{11}$$

If the $\tau_{TL}^f \ge \tau_{SSA}^f$, we sample a single reaction ($k_j^f = 1$) from the subset $\boldsymbol{R}_{SSA}^{f}$, such that j is the smallest integer satisfying

$$\sum_{i=1}^{j} a_i^f \geq \xi_2 \cdot \sum_{i \in \boldsymbol{R}_{SSA}^f} a_i^f \qquad (12)$$

If $\tau_{TL}^f < \tau_{SSA}^f$ then no reaction from the set $\boldsymbol{R}_{SSA}^f$ is executed.

The number of firings k_j^f of a reaction $R_j^f \in \boldsymbol{R}_{TL}^f$ is sampled from a Poisson distribution with mean $a_j^f \tau^f$

$$k_j^f = \boldsymbol{P}\left(a_j^f \tau^f\right), \quad \text{for all } j \in \boldsymbol{R}_{TL}^f \qquad (13)$$

Hybrid Solver as the macroscopic solver

Since macroscopic solver uses the slow-scale propensities as effective transition probabilities to evolve the slow network, employing the hybrid solver as the macroscopic solver mainly replaces the instantaneous value in Eqs. (9)-(13) with the slow scale propensities. We begin the implementation of the hybrid macrosolver by partitioning the slow reactions into a SSA subset, $\boldsymbol{R}_{SSA}^s$, and TL subset, $\boldsymbol{R}_{TL}^s$, based on the reactant populations. The time increment used to evolve the slow network is given by

$$\tau^s = \min\left\{\tau_{TL}^s, \tau_{SSA}^s\right\} \qquad (14)$$

where,

$$\tau_{SSA}^s = -\frac{\ln\left(\xi_1\right)}{\sum_{j \in \boldsymbol{R}_{SSA}^s} \overline{a_j^s}} \qquad (15)$$

$$\tau_{TL}^s = \min_{i \in \boldsymbol{S}} \left\{ \frac{r x_i}{\left| \sum_{j \in \boldsymbol{R}_{TL}^s;\, \upsilon_{ij}^s < 0} \upsilon_{ij}^s a_j^s\left(\mathbf{x^f}\right) \right|} \right\} \qquad (16)$$

We have used $i \in \boldsymbol{S}$ as a general condition, though in reality i corresponds to all species participating in the slow reactions, and thus some of the fast species might be excluded.

If $\tau_{SSA}^s \leq \tau_{TL}^s$, we sample a single reaction ($k_j^s = 1$) from the subset $\boldsymbol{R}_{SSA}^s$, such that j is the smallest integer satisfying

$$\sum_{i=1}^{j} \overline{a_i^s} \geq \xi_2 \cdot \sum_{i \in \boldsymbol{R}_{SSA}^s} \overline{a_i^s} \qquad (17)$$

If $\tau_{SSA}^s > \tau_{TL}^s$, then no reaction from the set $\boldsymbol{R}_{TL}^s$ is executed. The number of firings, k_j^s of a reaction $j \in \boldsymbol{R}_{TL}^s$ is sampled from a Poisson distribution with mean $\overline{a_j^s} \tau^s$

$$k_j^s = \boldsymbol{P}\left(\overline{a_j^s} \tau^s\right), \quad \text{for all } j \in \boldsymbol{R}_{TL}^s \qquad (18)$$

CONCLUSION

In this talk, we proposed a new hybrid, multiscale algorithm that uses hybrid solvers in a multiscale framework to give unprecedented speedup in simulating stiff, stochastic systems, without sacrificing simulation accuracy. The new relaxation criterion, which eliminates the need for complete quasi-equilibrium description, further enhances the efficiency of the hybrid multiscale technique. The novelty of the hybrid, multiscale Monte Carlo algorithm lies in its simultaneous handling of scale separation in two dimensions – time scale and population levels. More importantly, the algorithm can seamlessly switch between stiff/non-stiff and/or exact/coarse-grained solvers, depending on the scale separation encountered. The adaptive nature of the HyMSMC method allows it to overcome the limitations of other multiscale methods that make *à priori* assumptions about the scales in the system. The results presented during the talk will demonstrate the accuracy and efficiency of the new algorithm with the help of prototype and real biological examples. Additionally, we will also show how the efficiency and accuracy scale with scale separation in the network.

References

[1] D. T. Gillespie, Journal of Computational Physics **22**, 403 (1976); D. T. Gillespie, Journal of Physical Chemistry **81**, 2340 (1977).

[2] D. T. Gillespie, Journal of Chemical Physics **115** (4), 1716 (2001).

[3] A. Chatterjee, D. G. Vlachos, and M. A. Katsoulakis, Journal of Chemical Physics **122**, 024112 (2005).

[4] T. Tian and K. Burrage, Journal of Chemical Physics **121** (21), 10356 (2004).

[5] Y. Cao, Gillespie, D.T., Petzold, L.R., Journal of Chemical Physics **123** (5) (2005).

[6] M. A. Gibson and J. Bruck, Journal of Physical Chemistry A **104**, 1876 (2000).

[7] Y. Cao, D. T. Gillespie, and L. R. Petzold, Journal of Chemical Physics **122**, 014116 (2005).

[8] T. Cao, D. T. Gillespie, and L. R. Petzold, Journal of Computational Physics **206**, 395 (2005).

[9] A. Samant and D. G. Vlachos, J. Chem. Phys. **123** (14), 14114 (2005).

[10] E. L. Haseltine and J. B. Rawlings, Journal of Chemical Physics **117** (15), 6959 (2002).

[11] H. Salis and Y. Kaznessis, Journal of Chemical Physics **123**, 214106 (2005).

[12] H. Salis and Y. Kaznessis, Journal of Chemical Physics **122**, 054103 (2005).

[13] C. V. Rao and A. P. Arkin, Journal of Chemical Physics **118** (11), 4999 (2003).

[14] E. Weinan, D. Liu, and E. Vanden Eijnden, Journal of Chemical Physics **123**, 194107 (2005).

[15] R. E. Walpole, Myers, R.H. and Sharon, L.M., *Probability and Statistics for Engineers and Scientists*, 6th Edition ed. (Prentice Hall, 1998).