# Rejection-based Stochastic Simulation Algorithm for Biochemical Reaction Systems

Vo Hong Thanh[1]

**We present an efficient rejection-based stochastic simulation algorithm (RSSA) for biochemical reaction systems. RSSA is an exact simulation algorithm which improves the simulation by postponing and collapsing as much as possible the number of propensity computations. Propensity of each reaction in RSSA is varied in an interval denoted by a pair of the lower bound and upper bound values. RSSA will use propensity bounds to select the next reaction to fire in two steps: randomly select a candidate reaction, and then apply a rejection test on the candidate reaction to recover the exactness of the algorithm.**

***Keywords* — Computational biology, Stochastic simulation, Rejection-based stochastic simulation algorithm.**

## I. STOCHASTIC SIMULATION

We consider a well-mixed biochemical reaction system consisting of $n$ species $S_1 \dots S_n$ that can interact through $m$ reactions $R_1 \dots R_m$. The dynamical state at time $t$ is $X(t) = (X_1(t), \dots, X_n(t))$ where $X_i(t)$ denotes the current population of species $S_i$. The probability that a reaction $R_j$ fires in the next infinitesimal time $dt$ is quantified by $a_j(X(t))dt$ where $a_j$ is the propensity. The Gillespie's stochastic simulation algorithm (SSA) [1], in particular, is an exact method to realize possible state trajectories. Essentially, SSA selects the next reaction $R_j$ with a discrete probability $a_j/a_0$ where $a_0 = \sum_{j=1}^{m} a_j$. The reaction firing time $\tau$ is drawn from an exponential distribution with rate $a_0$. The state $X(t)$ is then updated accordingly to the selected reaction. At this new state, reactions have to update propensities to reflect changes in the state. A dependency graph [2] is often applied to decide which reactions actually need to be updated their propensities after a reaction fired.

The rejection-based stochastic simulation algorithm (RSSA) is an exact simulation with focusing on reducing the number of propensity updates. It does not need to compute propensities in most simulation loops. RSSA varies propensity of a reaction in an interval denoted by a pair of the lower bound and upper bound and makes use of these propensity bounds to select the next reaction firing [3].

## II. RSSA

For each reaction, the propensity $a_j$ is varied in an interval described by the lower bound $\underline{a_j}$ and upper bound $\overline{a_j}$. These propensity bounds are calculated by imposing a bound on the population of each species. For species $S_i$, its population $X_i(t)$ is bound to an interval $[\underline{X_i}, \overline{X_i}]$. The population bound for species $S_i$ can be chosen arbitrarily around current value $X_i(t)$ without affecting the correctness of the algorithm.

Given propensity bounds, RSSA selects the next reaction firing in two steps. A reaction $R_j$ is selected as a candidate with probability $\overline{a_j}/\overline{a_0}$ where $\overline{a_0} = \sum_{j=1}^{m} \overline{a_j}$. The selected candidate $R_j$ will enter a rejection test with success probability $a_j/\overline{a_j}$ to ensure it is selected with correct probability as SSA. The validation avoids computing exact propensity $a_j$ as much as possible by making use the fact that $\underline{a_j}/\overline{a_j} \le a_j/\overline{a_j}$. Thus, a random number $r \sim U(0,1)$ is drawn and checks whether $r \le \underline{a_j}/\overline{a_j}$. If this test succeeds, the candidate $R_j$ is accepted immediately because of $r \le \underline{a_j}/\overline{a_j} \le a_j/\overline{a_j}$. If it fails, the exact value $a_j$ is indeed computed using the current state *X(t)*, and $r$ is tested against $a_j/\overline{a_j}$. If $R_j$ is accepted, its firing time $\tau$ is drawn from an *Erlang(k,* $\overline{a_0}$*)* distribution with parameter *k* is the number of trials until the reaction is accepted. In case $R_j$ is rejected, the selection procedure is repeated.

After a reaction is accepted, the state *X(t)* is updated. RSSA will check whether the population of each species is still compatible with its bound, i.e., $\underline{X_i} \le X_i \le \overline{X_i}$ for all species $S_i$. If this happens, the next simulation iteration is performed without recomputing the propensities. However, in the uncommon case in which the condition fails, a new population bound for species is defined, and propensity bounds are derived. The details and the proof for the exactness of RSSA can be seen in [3].

## III. CONCLUSION

We have presented an exact stochastic simulation algorithm, called RSSA, which accelerates the simulation by reducing the need of propensity updates. RSSA exploits the propensity bounds of reactions and a rejection-based mechanism to selects the next reaction firing.

## REFERENCES

[1] Daniel T. Gillespie (1976). *J. Comp. Phys.* 22, 4 (1976), 403–434
[2] Michael Gibson and Jehoshua Bruck (2000). *J. Phys. Chem.* 104, 9, 1876–1889.
[3] Vo Hong Thanh (2013). *On efficient stochastic simulation of biochemical reaction systems*. Ph.D. Dissertation. University of Trento, Italy.

[1]Microsft Research - University of Trento Centre for Computational and System Biology. E-mail: vo@cosbi.eu