Auger et al.,
J. Chem. Phys. 2006

R-Leaping

Accelerating the SSA by Reaction leaps

Petros Koumoutsakos

WITH : A. Auger, P. Chatelain

CSE Lab
Computational Science & Engineering Laboratory
http://www.cse-lab.ethz.ch

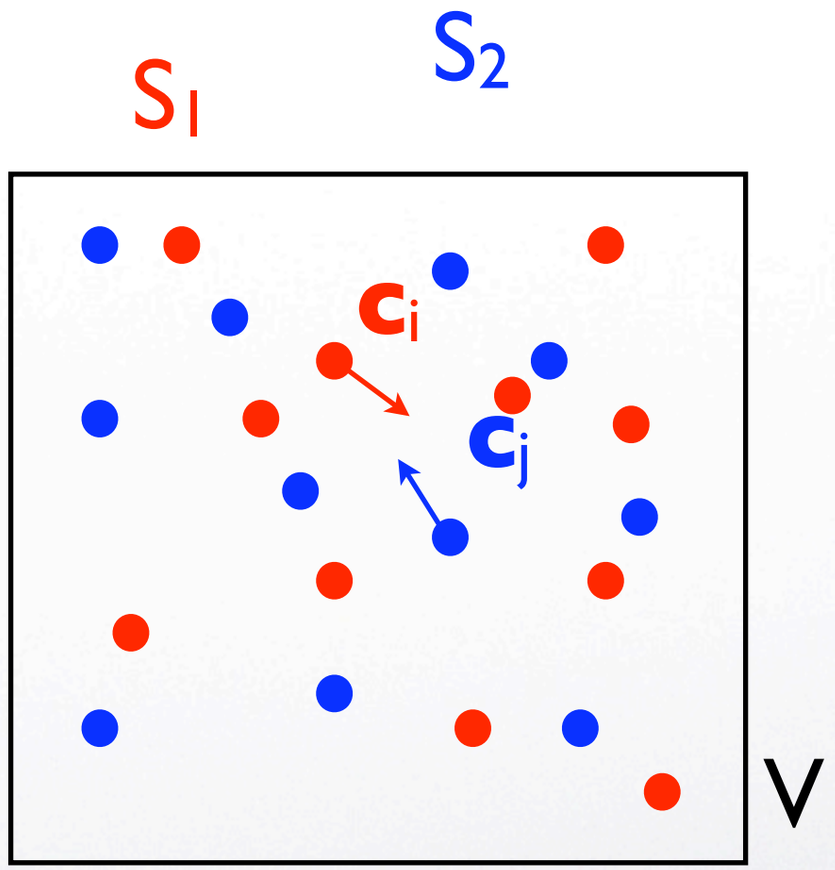
ETH
Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Outline

- Stochastic simulations
  - Exact -> SSA
  - Approximate -> τ-leaping (taph - leaping)
  - Approximate -> R-leaping
- Comparisons and Results
- Outlook

Chemical kinetics : Set-up

- Well stirred reaction volume V
- N different species $S_1, S_2, \ldots, S_N$ in numbers $X_1, X_2, \ldots, X_N$
- random collisions and reactions through M channels $R_1, R_2, \ldots, R_M$
- Experiment length T



Stochastic simulation: SSA

Gillespie,
J. Comp. Phys. 1977

- For M reactions, time until any reaction

$$\tau \sim \mathcal{E}(1/a_0) \qquad a_0 = \sum_{j=1}^{M} a_j$$

- Reaction index : point-wise distribution

$$p(j = l) = \frac{a_l}{a_0}$$

- One timestep:
  - Sample τ
  - Sample the index j
  - Update the X_i, $t=t+\tau$
- The SSA simulates every reaction event !
-

Stochastic simulation: acceleration

- **SSA** : exact but slow
- **τ leaping** : several reaction events over one time step,

Gillespie,
J. Chem. Phys. 2001

- **Assumption** : reaction propensities a_j remain essentially constant over **τ**, in spite of several firings

- Over this given **τ**, the number of reaction firings K^P_j is governed by a Poisson distribution

$$K_j^{\mathcal{P}} \sim \mathcal{P}(a_j \tau)$$
$$\mathbf{X}(t+\tau) = \mathbf{X}(t) + \sum_{j=1}^{M} K_j^{\mathcal{P}} \boldsymbol{\nu}_j.$$

Cost ~ M Poisson samplings

τ leaping Consequences

- τ leaping : Can generate negative populations
- Binomial τ leaping : Approximate the unbounded Poisson distributions with Binomial ones

Tian & Burrage,
J. Chem. Phys. 2004

- Modified τ leaping

Chatterjee et al.,
J. Chem. Phys. 2005

  - Critical reactions, i.e. those likely to drive some populations negative, handled by SSA
  - Other reactions advanced by τ leaping

Cao et al.,
J. Chem. Phys. 2005

R-leaping : Accelerate SSA by reaction leaps

Auger et al.,
J. Chem. Phys. 2006

R-leaping : Accelerate SSA by reaction leaps

Auger et al.,
J. Chem. Phys. 2006

- Leaps are in **prescribed number of reaction firings L across all reaction channels**

R-leaping : Accelerate SSA by reaction leaps

Auger et al.,
J. Chem. Phys. 2006

Leaps are in **prescribed number of reaction firings L across all reaction channels**

- Time increment τ_L is Gamma-distributed $\tau_L \sim \Gamma(L, 1/a_0(\mathbf{x}))$
- In this interval we will have K_m firings of channel R_m
- with : $\sum_{m=1}^{M} K_m = L$

R-leaping : Accelerate SSA by reaction leaps

Auger et al.,
J. Chem. Phys. 2006

Leaps are in **prescribed number of reaction firings L across all reaction channels**

- Time increment τ_L is Gamma-distributed $\tau_L \sim \Gamma(L, 1/a_0(\mathbf{x}))$
- In this interval we will have K_m firings of channel R_m
- with : $\sum_{m=1}^{M} K_m = L$
- In R-leaping, as in SSA, the index j of every firing obeys a point-wise distribution

$$P(j = l) = \frac{a_l(\mathbf{x})}{a_0(\mathbf{x})} \quad \text{for } l = 1, \ldots, M.$$

R-leaping : One step

Auger et al.,
J. Chem. Phys. 2006

- Define L

$$\tau_L \sim \Gamma(L, 1/a_0(\mathbf{x}))$$

- Sample the index j

$$P(j = l) = \frac{a_l(\mathbf{x})}{a_0(\mathbf{x})} \quad \text{for } l = 1, \ldots, L.$$

- Number of reactions for channel m

$$K_m = \sum_{l=1}^{L} \delta_{l,m}$$

- Update species and time :

$$\mathbf{X}(t + \tau_L) = \mathbf{X}(t) + \sum_{j=1}^{M} K_j \boldsymbol{\nu}_j$$

R-leaping : Accelerate SSA by reaction leaps

- L firings distributed across M reaction channels
  - In τ leaping: K^P_j are independent Poisson variables.
  - In R-leaping, K_j are not independent.
- L as a control parameter
  - System can be brought to a desired state X
  - Time is not a-priori specified
  - New approaches to controlling negative species

R-leaping : How to Sample the the **M** K_j

R_0 Algorithm

$$p(j = l) = \frac{a_l}{a_0}$$

Ro-sampling scales with L and, in particular when compared with **τ** -leaping that scales with M, the method is inefficient for large leap sizes, L ≫ M.

R-leaping : How to Sample the the **M** K_j

R_0 **Algorithm**

- Pointwise Sampling of **L** *independent* reaction indices

$$p(j = l) = \frac{a_l}{a_0}$$

- Simple BUT scales with L - close to the work load of SSA!

Ro-sampling scales with L and, in particular when compared with τ -leaping that scales with M, the method is inefficient for large leap sizes, L ≫ M.

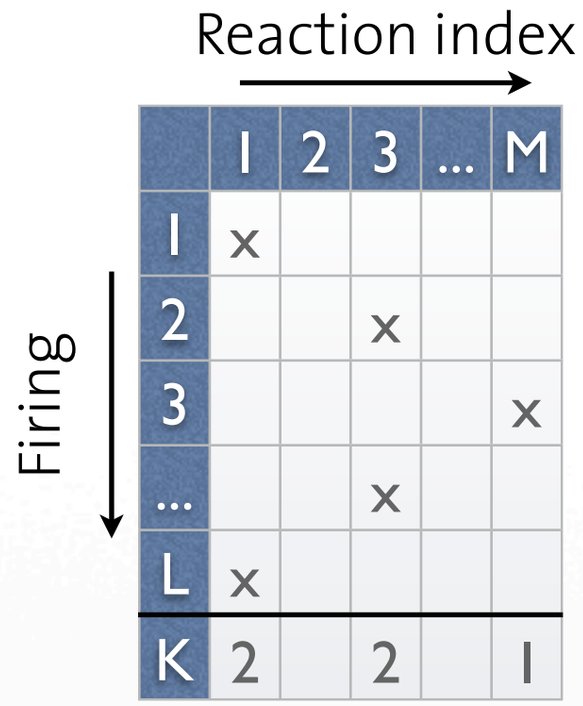
R-leaping : How to Sample the the **M** K_j

R_0 **Algorithm**

- Pointwise Sampling of **L** *independent* reaction indices

$$p(j = l) = \frac{a_l}{a_0}$$

- Simple BUT scales with L - close to the work load of SSA!



Ro-sampling scales with L and, in particular when compared with τ -leaping that scales with M, the method is inefficient for large leap sizes, L ≫ M.

R-Leaping Theorem

The distribution of K_1 is a binomial distribution :

$$\mathcal{B}(L, a_1(\mathbf{x})/a_0(\mathbf{x}))$$

and for every $m \in \{2, \ldots, M\}$ the conditional distribution of K_m

given the event $\{(K_1, \ldots, K_{m-1}) = (k_1, \ldots, k_{m-1})\}$ is

$$K_m \sim \mathcal{B}\left(L - \sum_{i=1}^{m-1} k_i, \frac{a_m(\mathbf{x})}{a_0(\mathbf{x}) - \sum_{i=1}^{m-1} a_i(\mathbf{x})}\right).$$

This result is invariant under any permutation of the indices

Lemmas for R-leaping Theorem

Lemma I

For every $m = 1, \ldots, M$ the random variables K_m follow a Binomial distribution with parameters L and $a_m(\mathbf{x})/a_0(\mathbf{x})$

$$K_m \sim \mathcal{B}(L, a_m(\mathbf{x})/a_0(\mathbf{x}))$$

PROOF :

Assume that the indices $(l = 1, \ldots, L)$ have been drawn from point-wise distributions.

Fix an integer $m \in [1, M]$

For each index $(l = 1, \ldots, L)$ there are only two possible outcomes

- $l = m$ with probability $a_m(\mathbf{x})/a_0(\mathbf{x})$
- or not with probability $1 - a_m(\mathbf{x})/a_0(\mathbf{x})$

Each event $l = m$ is then Bernoulli distributed with probability $a_m(\mathbf{x})/a_0(\mathbf{x})$

Since K_m is the number of successes in these L independent Bernoulli sampling, it will follow the distribution $\mathcal{B}(L, a_m(\mathbf{x})/a_0(\mathbf{x}))$

Lemmas for R-leaping Theorem

Lemma II

The following holds

$$P(l = 2|l > 1) = \frac{a_2(\mathbf{x})}{a_0(\mathbf{x}) - a_1(\mathbf{x})}$$

$$P(l \neq 2|l > 1) = 1 - \frac{a_2(\mathbf{x})}{a_0(\mathbf{x}) - a_1(\mathbf{x})}$$

R-leaping : How to Sample the the **M** K_j

R_0 `Algorithm`

$$p(j = l) = \frac{a_l}{a_0}$$

Ro-sampling scales with L and, in particular when compared with **τ** -leaping that scales with M, the method is inefficient for large leap sizes, L ≫ M.

R-leaping : How to Sample the the **M** K_j

R_0 `Algorithm`

- Pointwise Sampling of **L** *independent* reaction indices

$$p(j = l) = \frac{a_l}{a_0}$$

  - Simple BUT scales with L - close to the work load of SSA!

Ro-sampling scales with L and, in particular when compared with τ -leaping that scales with M, the method is inefficient for large leap sizes, L ≫ M.

R-leaping : How to Sample the the **M** K_j

R_0 Algorithm

- Pointwise Sampling of **L** *independent* reaction indices

$$p(j = l) = \frac{a_l}{a_0}$$

  - Simple BUT scales with L - close to the work load of SSA!

Ro-sampling scales with L and, in particular when compared with τ -leaping that scales with M, the method is inefficient for large leap sizes, L ≫ M.

R_1 Algorithm

- Sampling **M** *correlated* binomial variables

$$\mathcal{B}(L, a_j/a_0)$$

  - Create correlations with conditional distributions

$$\text{If } K_i = k_i,\ \forall i < m,$$

$$K_m \sim \mathcal{B}\left(L - \sum_{i=1}^{m-1} k_i,\ \frac{a_m}{a_0 - \sum_{i=1}^{m-1} a_i}\right)$$

R-leaping : How to Sample the the **M** K_j

R_0 Algorithm

- Pointwise Sampling of **L** *independent* reaction indices

$$p(j = l) = \frac{a_l}{a_0}$$

  - Simple BUT scales with L - close to the work load of SSA!

Reaction index

Firing	1	2	3	...	M
1	x				
2			x		
3					x
...			x		
L	x				
K	2		2		1

Ro-sampling scales with L and, in particular when compared with τ -leaping that scales with M, the method is inefficient for large leap sizes, L ≫ M.

R_1 Algorithm

- Sampling **M** *correlated* binomial variables

$$\mathcal{B}(L, a_j/a_0)$$

  - Create correlations with conditional distributions

$$\text{If } K_i = k_i,\ \forall i < m,$$

$$K_m \sim \mathcal{B}\left(L - \sum_{i=1}^{m-1} k_i,\ \frac{a_m}{a_0 - \sum_{i=1}^{m-1} a_i}\right)$$

R-leaping : How to Sample the the **M** K_j

R_0 Algorithm

- Pointwise Sampling of **L** *independent* reaction indices

$$p(j = l) = \frac{a_l}{a_0}$$

  - Simple BUT scales with L - close to the work load of SSA!

Reaction index

Firing	1	2	3	...	M
1	x				
2			x		
3					x
...			x		
L	x				
K	2		2		1

Ro-sampling scales with L and, in particular when compared with τ -leaping that scales with M, the method is inefficient for large leap sizes, L ≫ M.

R_1 Algorithm

- Sampling **M** *correlated* binomial variables

$$\mathcal{B}(L, a_j/a_0)$$

  - Create correlations with conditional distributions

$$\text{If } K_i = k_i,\ \forall i < m,$$

$$K_m \sim \mathcal{B}\left(L - \sum_{i=1}^{m-1} k_i,\ \frac{a_m}{a_0 - \sum_{i=1}^{m-1} a_i}\right)$$

Reaction index

Firing	1	2	3	...	M
1	x				
2			x		
3					x
...			x		
L	x				
K	2		2		1

R-leaping : Sorting for efficiency

- Sampling the **M** K_j efficiently

$$\text{If } K_i = k_i,\ \forall i < m,\ K_m \sim \mathcal{B}\left(L - \sum_{i=1}^{m-1} k_i,\ \frac{a_m}{a_0 - \sum_{i=1}^{m-1} a_i}\right)$$

  - When $\sum_{i=1}^{m-1} k_i = L$, sampling is done!
  - Minimize the average m by a permutation of the indices, such that $a_{j'}$ is decreasing

    - E.g. $a_M > a_3 > a_1 >> \cdots$

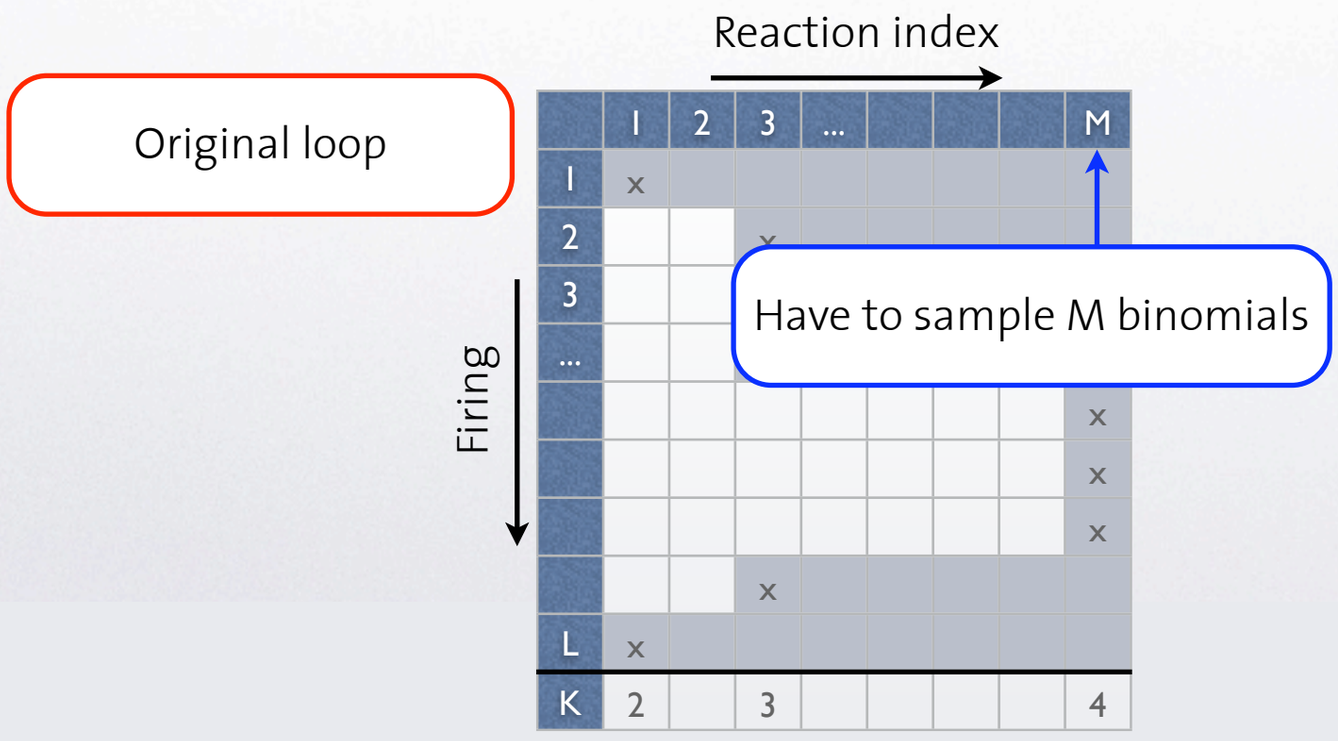
R-leaping : Sorting for efficiency

- Sampling the **M** K_j efficiently

$$\text{If } K_i = k_i,\ \forall i < m,\ K_m \sim \mathcal{B}\left(L - \sum_{i=1}^{m-1} k_i,\ \frac{a_m}{a_0 - \sum_{i=1}^{m-1} a_i}\right)$$

  - When $\sum_{i=1}^{m-1} k_i = L$, sampling is done!
  - Minimize the average m by a permutation of the indices, such that $a_{j'}$ is decreasing

    - E.g. $a_M > a_3 > a_1 >> \cdots$



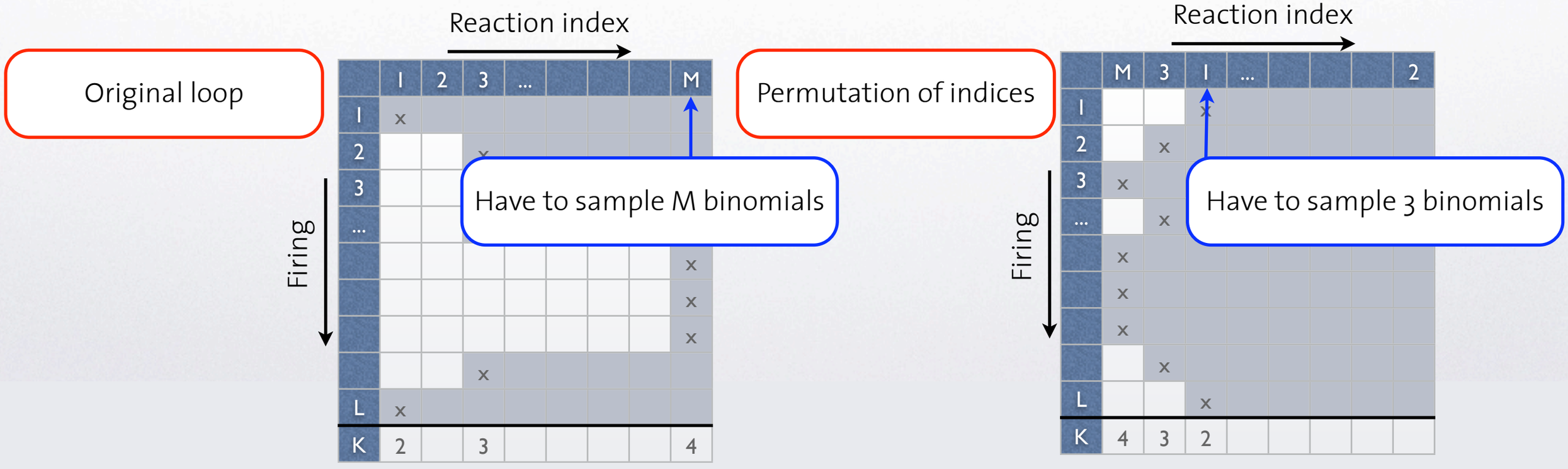
R-leaping : Sorting for efficiency

- Sampling the **M** K_j efficiently

$$\text{If } K_i = k_i,\ \forall i < m,\ K_m \sim \mathcal{B}\left(L - \sum_{i=1}^{m-1} k_i,\ \frac{a_m}{a_0 - \sum_{i=1}^{m-1} a_i}\right)$$

  - When $\sum_{i=1}^{m-1} k_i = L$, sampling is done!
  - Minimize the average m by a permutation of the indices, such that $a_{j'}$ is decreasing

    - E.g. $a_M > a_3 > a_1 >> \cdots$

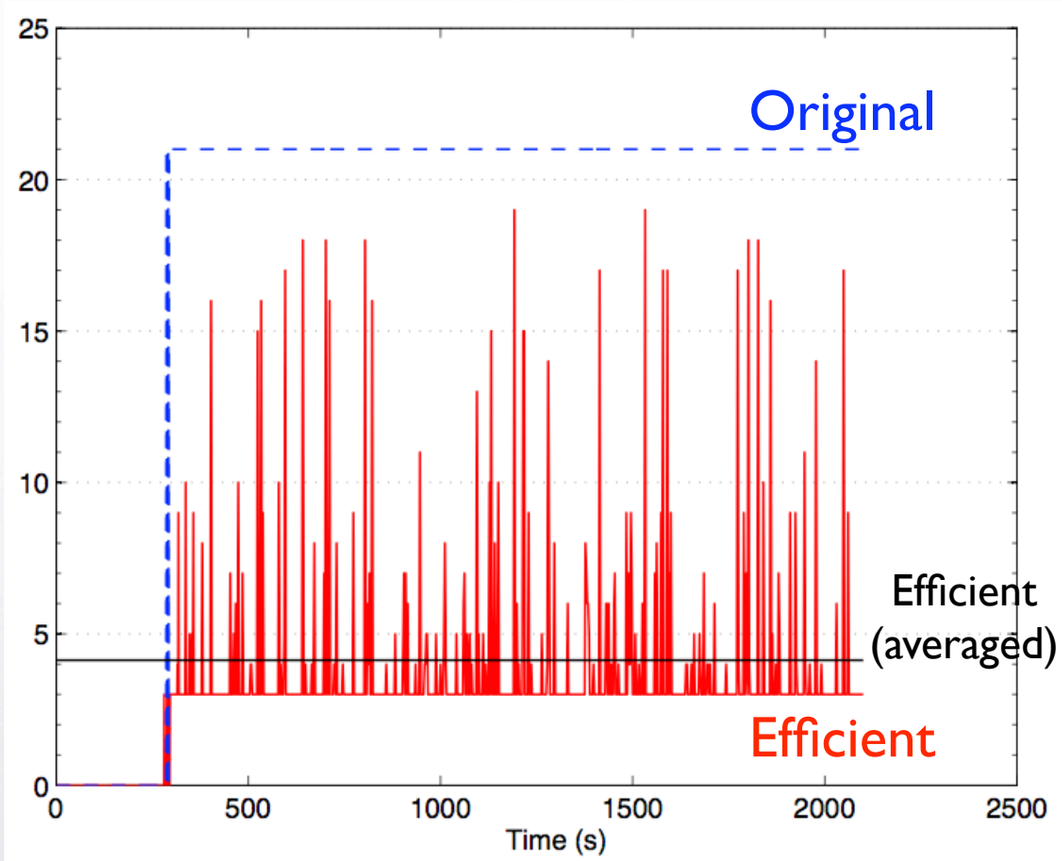


R-leaping : Efficient Sampling

- Sampling the **M** K_j efficiently
  - **M** can be large (~10^2) for bio-chemical systems!
  - Efficient sampling effectively loops over a fraction of **M**.

- The larger the system, the bigger the payoff.
- The more disparate the reaction rates are, the smaller the fraction.
- Price to pay: carry out re-ordering often enough (cheap!)

Number of binomial samples per time step
LacYLacZ activities in E. Coli., M=22



Stochastic simulation: R-leaping

Stochastic simulation: R-leaping

- Controlling the leap approximation

Stochastic simulation: R-leaping

- Controlling the leap approximation
  - All three methods of τ leaping are transposable to R-leaping
    - Absolute change of a_j

Stochastic simulation: R-leaping

- Controlling the leap approximation
  - All three methods of τ leaping are transposable to R-leaping
    - Absolute change of a_j
    - Relative change of a_j
    - Relative change of a_j but efficiently through the relative changes in populations

Approximation control : τ leaping

- The L>1 fired reactions are changing the propensities
- τ "small enough" or bound the propensity change
  - want to impose, $\forall j\ ,\ \Delta a_j < \epsilon a_j$ to get τ estimate
  - Only possible in a probabilistic sense

$$\forall j\ ,\ \mathbb{E}(\Delta a_j) < \epsilon a_j$$

- Taylor expansion and truncation at first order gives bounds on τ

Gillespie & Petzold,
J. Chem. Phys. 2003

$$\forall j\ ,\ \mathrm{var}(\Delta a_j) < \epsilon a_j$$

τ leaping : Bounds on propensity changes

- Use the absolute change $\forall j\ ,\ \Delta a_j < \epsilon a_0$
  - but if a_0 is large, slow reactions can see their propensities go through huge changes!
- Use the relative changes $\forall j\ ,\ \Delta a_j < \epsilon a_j$
  - Solves accuracy but...
- the computation of $\mathbb{E}(\Delta a_j)$ and $\mathrm{var}(\Delta a_j)$ involves the determination of influences of reaction firings over all the propensities

$$f_{jm}(\mathbf{x}) = \sum_{l=1}^{N} \frac{\partial a_j(\mathbf{x})}{\partial x_l} \nu_{lm}$$

= how much one firing of R_l changes a_j

- M^2 entries! Sparse but still heavy...

τ leaping : Bounds on the propensity changes

- Use the relative changes in populations
  - For a first order reaction, $S_i \longrightarrow \text{products}$ $a_j(\mathbf{x}) = c_j x_i$ we have

$$\frac{\Delta a_j}{a_j} = \frac{\Delta x_i}{x_i}$$

Cao et al.,
J. Chem. Phys. 2006

- Bound on relative changes is satisfied!
- Similar relations for second, third order
- Leap control scales with **N**, **M**. No more large sparse (**M**x**M**) matrix involved

Stochastic simulation: R-leaping

Stochastic simulation: R-leaping

- Controlling negative species with **L** as a leap parameter

Stochastic simulation: R-leaping

- Controlling negative species with **L** as a leap parameter
  - Bound **L** by maximum number of firings $\mathbf{L_j}$ allowed by reactants
    - Strict control of negative species

$$L < L_j$$

Stochastic simulation: R-leaping

- Controlling negative species with **L** as a leap parameter
  - Bound **L** by maximum number of firings $\mathbf{L_j}$ allowed by reactants
    - Strict control of negative species

$$L < L_j$$

Too strict!

Stochastic simulation: R-leaping

- Controlling negative species with **L** as a leap parameter
  - Bound **L** by maximum number of firings $\mathbf{L_j}$ allowed by reactants

$$L < L_j$$

    - Strict control of negative species

Too strict!

  - Bound the expected number of firings for reaction j by $\mathbf{L_j}$

$$L\frac{a_j}{a_0} < L_j$$

- Introduces negative species

Stochastic simulation: R-leaping

- Controlling negative species with **L** as a leap parameter
  - Bound **L** by maximum number of firings $\mathbf{L_j}$ allowed by reactants

$$L < L_j$$

    - Strict control of negative species

Too strict!

  - Bound the expected number of firings for reaction j by $\mathbf{L_j}$

$$L\frac{a_j}{a_0} < L_j$$

    - Introduces negative species

Too lax!

Stochastic simulation: R-leaping

- Controlling negative species with **L** as a leap parameter
  - Bound **L** by maximum number of firings $\mathbf{L_j}$ allowed by reactants

$$L < L_j$$

    - Strict control of negative species

Too strict!

  - Bound the expected number of firings for reaction j by $\mathbf{L_j}$

$$L\frac{a_j}{a_0} < L_j$$

    - Introduces negative species

Too lax!

- Introduce trade-off
  - With parameter θ
  - Allows negative species at a controlled frequency

$$L \leq (1-\theta)L_j + \theta\frac{a_0}{a_j}L_j$$

Stochastic simulation: R-leaping

- Controlling negative species with **L** as a leap parameter
  - Bound **L** by maximum number of firings $\mathbf{L_j}$ allowed by reactants

$$L < L_j$$

    - Strict control of negative species

Too strict!

  - Bound the expected number of firings for reaction j by $\mathbf{L_j}$

$$L\frac{a_j}{a_0} < L_j$$

    - Introduces negative species

Too lax!

- Introduce trade-off
  - With parameter θ
  - Allows negative species at a controlled frequency

$$L \leq (1-\theta)L_j + \theta\frac{a_0}{a_j}L_j$$

Tunable compromise between efficiency and accuracy

Results

- Accuracy and complexity
  - Decaying-dimerizing system
  - no negative species

$$S_1 \xrightarrow{c_1} 0, \quad S_1 + S_1 \xrightarrow{c_2} S_2, \quad S_2 \xrightarrow{c_3} S_1 + S_1, \quad S_2 \xrightarrow{c_4} S_3$$

$$c_1 = 1, \quad c_2 = 0.002, \quad c_3 = 0.5, \quad c_4 = 0.04 \ .$$

$$X_1(0) = 4150, \ X_2(0) = 39565, \ X_3(0) = 3445$$

Results

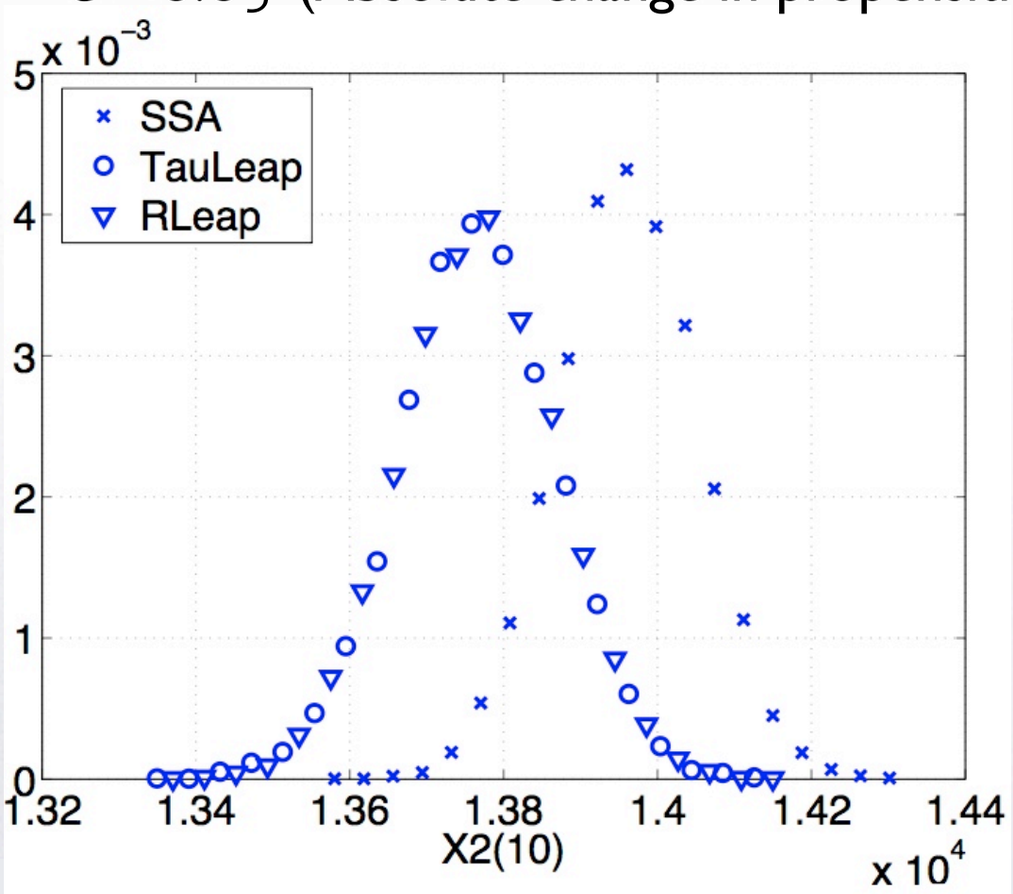
- Accuracy and complexity
  - Decaying-dimerizing system
  - no negative species

$$S_1 \xrightarrow{c_1} 0, \quad S_1 + S_1 \xrightarrow{c_2} S_2, \quad S_2 \xrightarrow{c_3} S_1 + S_1, \quad S_2 \xrightarrow{c_4} S_3$$

$$c_1 = 1, \quad c_2 = 0.002, \quad c_3 = 0.5, \quad c_4 = 0.04 \ .$$

$$X_1(0) = 4150, \ X_2(0) = 39565, \ X_3(0) = 3445$$

ε = 0.05 (Absolute change in propensities)



Histogram of X_2 at t = 10

Results

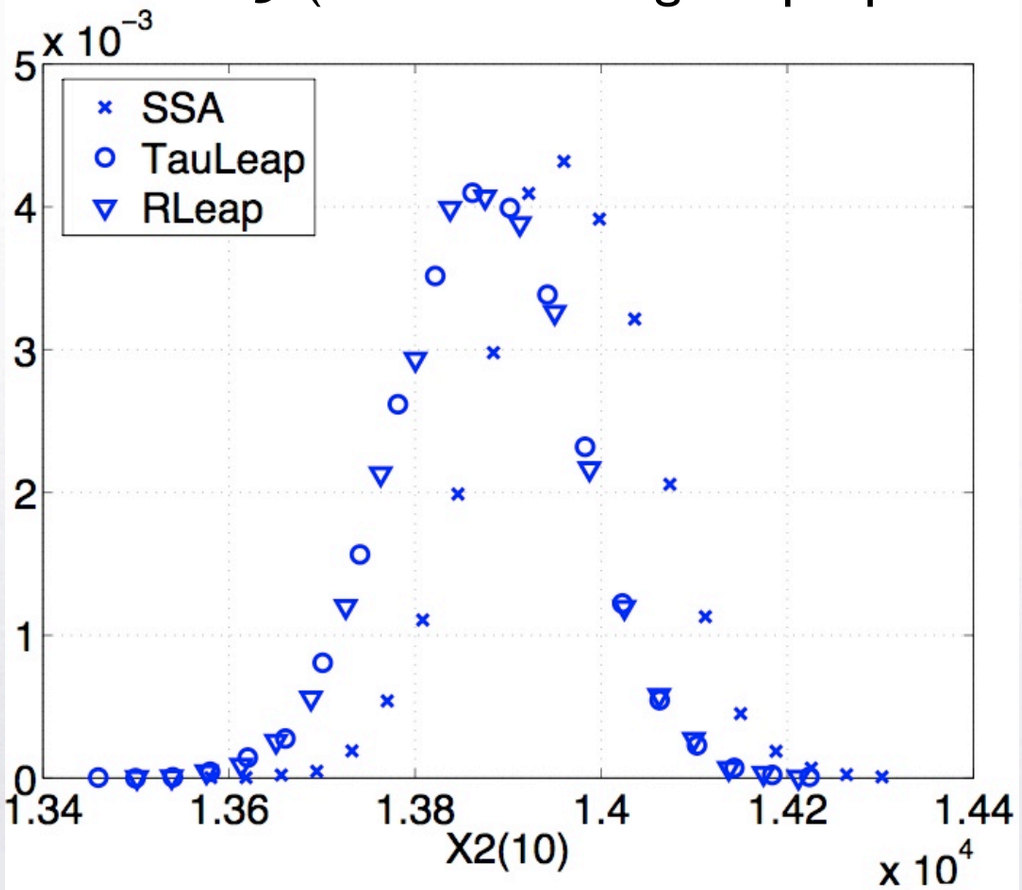
- Accuracy and complexity
  - Decaying-dimerizing system
  - no negative species

$$S_1 \xrightarrow{c_1} 0, \quad S_1 + S_1 \xrightarrow{c_2} S_2, \quad S_2 \xrightarrow{c_3} S_1 + S_1, \quad S_2 \xrightarrow{c_4} S_3$$

$$c_1 = 1, \quad c_2 = 0.002, \quad c_3 = 0.5, \quad c_4 = 0.04 \ .$$

$$X_1(0) = 4150, \ X_2(0) = 39565, \ X_3(0) = 3445$$

ε = 0.03 (Absolute change in propensities)



Histogram of X_2 at t = 10

Results

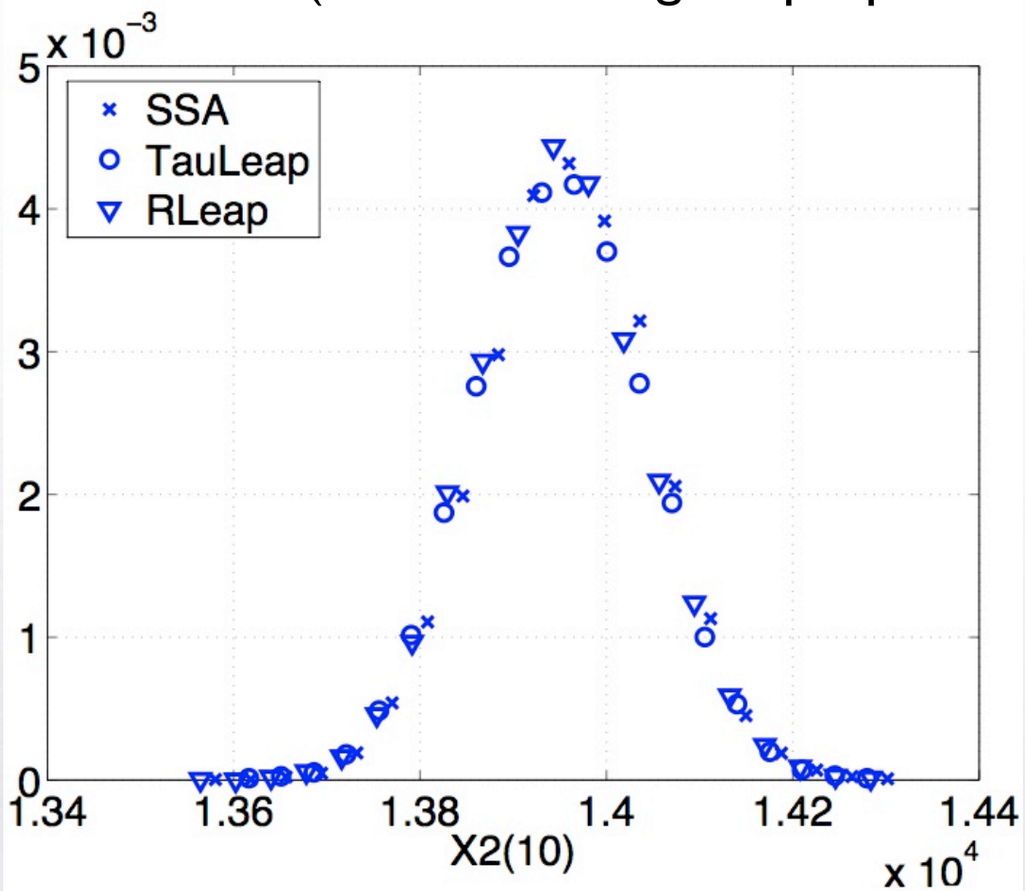
- Accuracy and complexity
  - Decaying-dimerizing system
  - no negative species

$$S_1 \xrightarrow{c_1} 0, \quad S_1 + S_1 \xrightarrow{c_2} S_2, \quad S_2 \xrightarrow{c_3} S_1 + S_1, \quad S_2 \xrightarrow{c_4} S_3$$

$$c_1 = 1, \quad c_2 = 0.002, \quad c_3 = 0.5, \quad c_4 = 0.04 \ .$$

$$X_1(0) = 4150, \ X_2(0) = 39565, \ X_3(0) = 3445$$

ε = 0.01 (Absolute change in propensities)



Histogram of X_2 at t = 10

Results

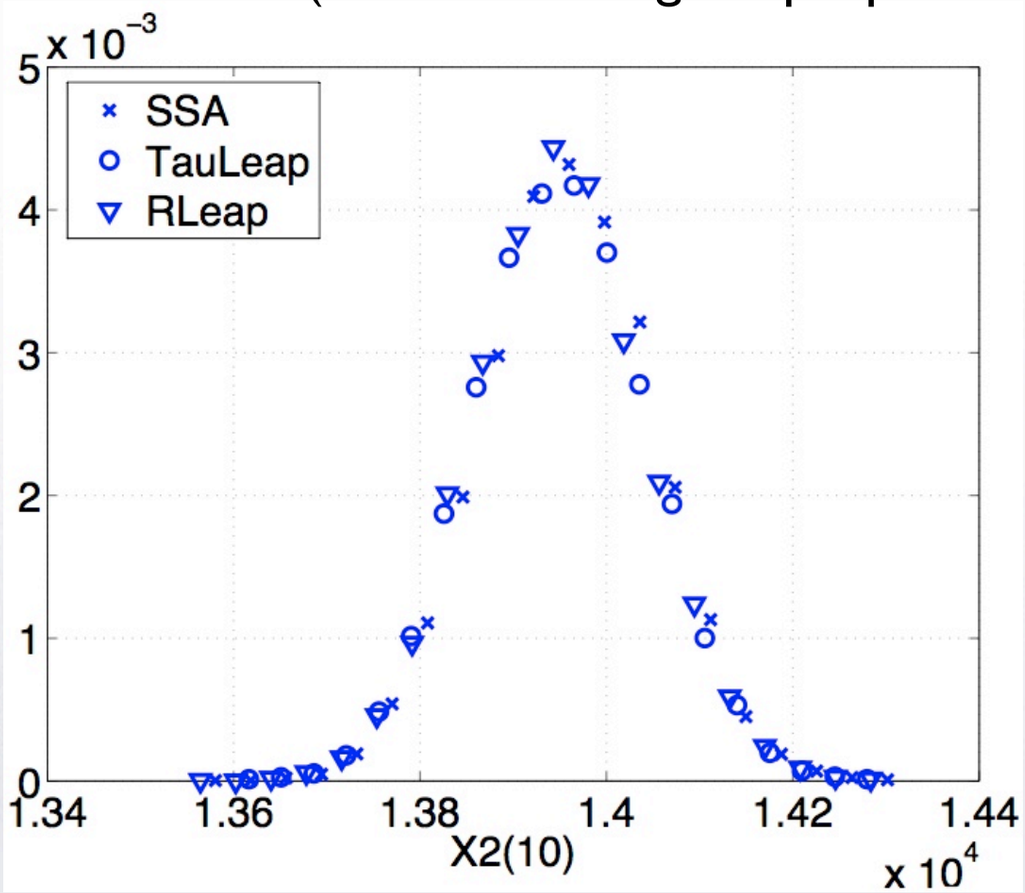
- Accuracy and complexity
  - Decaying-dimerizing system
  - no negative species

$$S_1 \xrightarrow{c_1} 0, \quad S_1 + S_1 \xrightarrow{c_2} S_2, \quad S_2 \xrightarrow{c_3} S_1 + S_1, \quad S_2 \xrightarrow{c_4} S_3$$

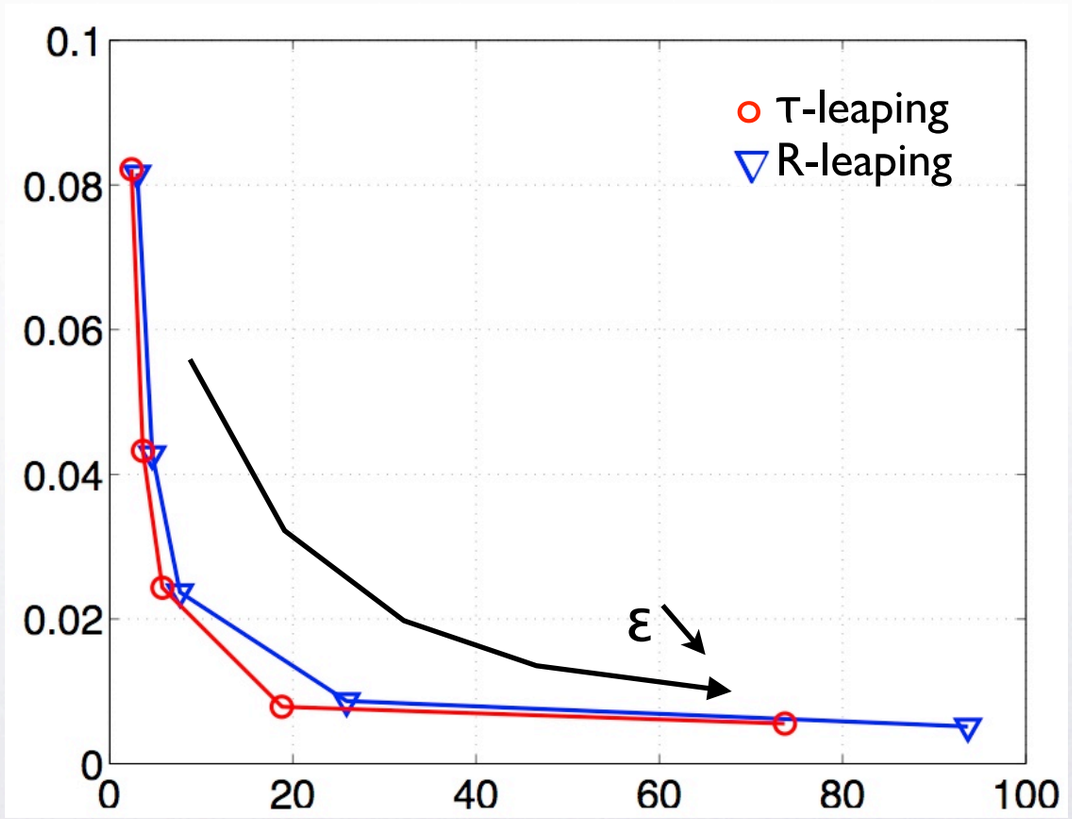
$$c_1 = 1, \quad c_2 = 0.002, \quad c_3 = 0.5, \quad c_4 = 0.04 \ .$$

$$X_1(0) = 4150, \ X_2(0) = 39565, \ X_3(0) = 3445$$

ε = 0.01 (Absolute change in propensities)



Histogram of X_2 at t = 10



Histogram error vs. CPU time

Results

- LacZ/LacY genes expression and enzymatic/ transport activities of LacZ/LacY proteins in E. Coli

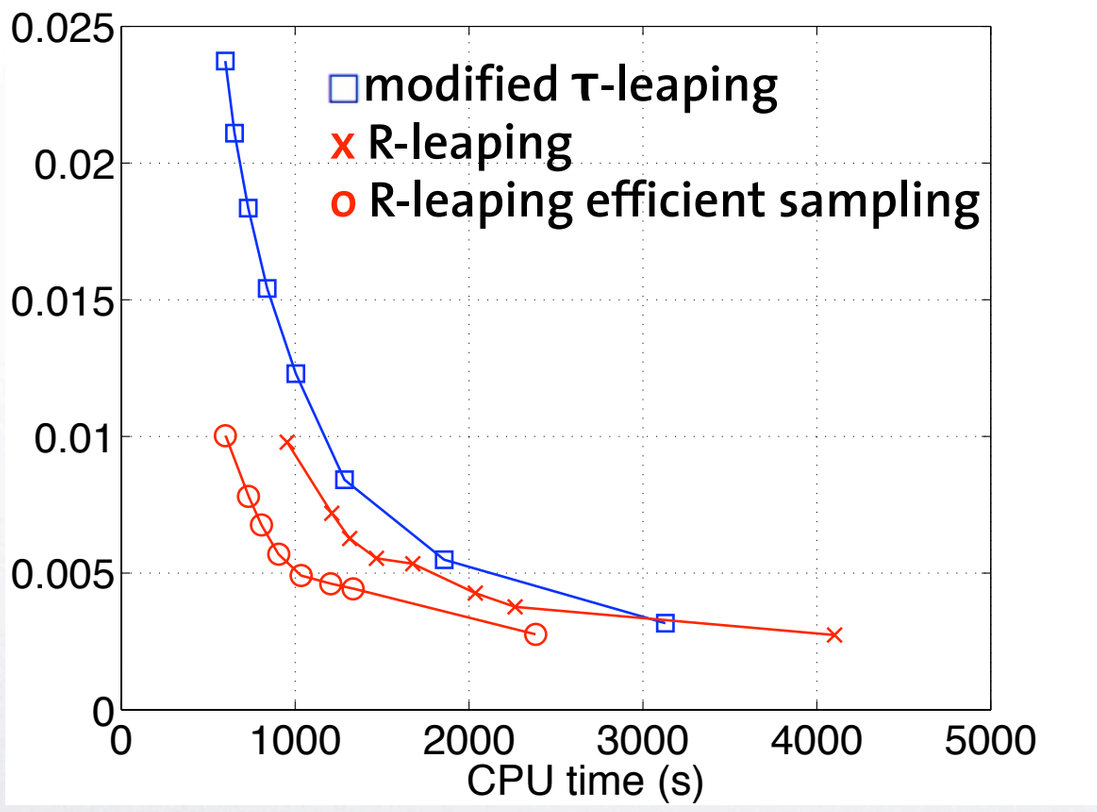
Kierzek, Bioiformatics 2002

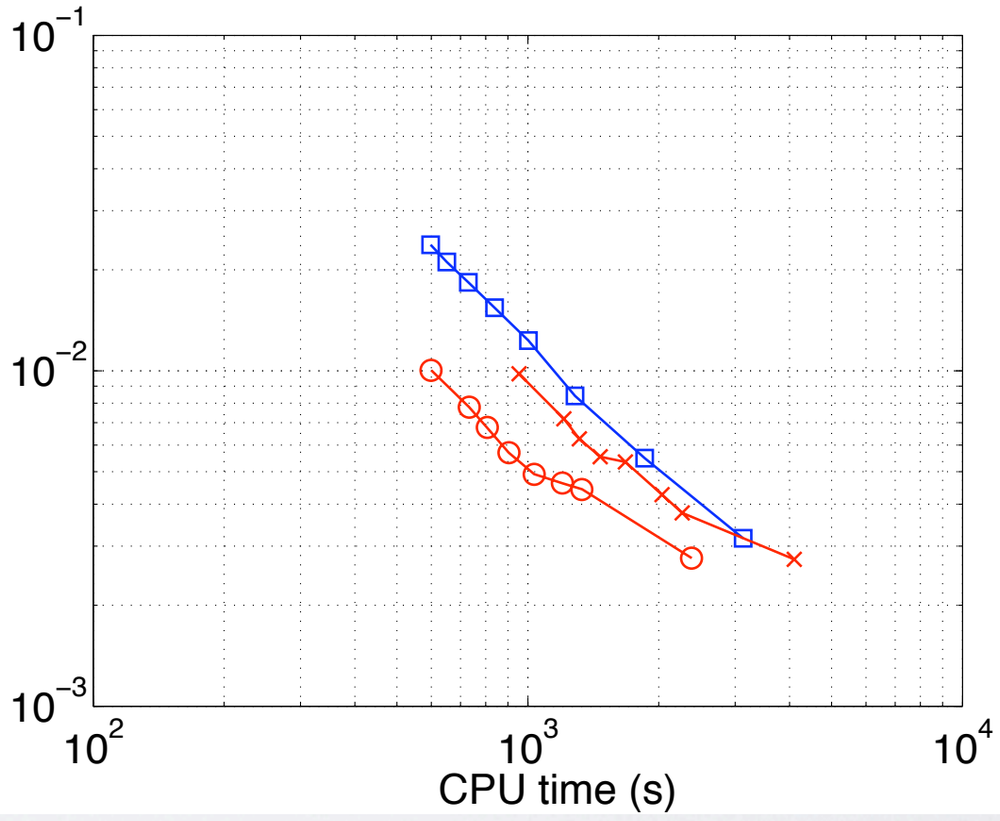
  - Moderately large system (M = 22)
  - Disparate rates
  - Scarce reactants and negative species

	Reaction Channel	Reaction rate
R_1	PLac + RNAP → PLacRNAP	0.17
R_2	PLacRNAP → PLac + RNAP	10
R_3	PLacRNAP → TrLacZ1	1
R_4	TrLacZ1 → RbsLacZ + PLac + TrLacZ2	1
R_5	TrLacZ2 → TrLacY2	0.015
R_6	TrLacY1 → RbsLacY + TrLacY2	1
R_7	TrLacY2 → RNAP	0.36
R_8	Ribosome + RbsLacZ → RbsRibosomeLacZ	0.17
R_9	Ribosome + RbsLacY → RbsRibosomeLacY	0.17
R_{10}	RbsRibosomeLacZ → Ribosome + RbsLacZ	0.45
R_{11}	RbsRibosomeLacY → Ribosome + RbsLacY	0.45
R_{12}	RbsRibosomeLacZ → TrRbsLacZ + RbsLacZ	0.4
R_{13}	RbsRibosomeLacY → TrRbsLacY + RbsLacY	0.4
R_{14}	TrRbsLacZ → LacZ	0.015
R_{15}	TrRbsLacY → LacY	0.036
R_{16}	LacZ → dgrLacZ	$6.42\text{x}10^{-5}$
R_{17}	LacY → dgrLacY	$6.42\text{x}10^{-5}$
R_{18}	RbsLacZ → dgrRbsLacZ	0.3
R_{19}	RbsLacY → dgrRbsLacY	0.3
R_{20}	LacZ + lactose → LacZlactose	$9.52\text{x}10^{-5}$
R_{21}	LacZlactose → product + LacZ	431
R_{22}	LacY → lactose + LacY	14

Results

- LacZ/LacY genes expression and enzymatic/ transport activities of LacZ/LacY proteins in E. Coli
  - Histogram errors vs CPU time





- Efficient sampling offers factor 2 in speed w.r.t. modified τ-leaping!

Summary

- **R-leaping**, an accelerated stochastic algorithm that is complementary to existing **τ-leaping** algorithms

- Efficient binomial sampling offers computational savings for large **systems with disparate rates**
  - Efficient sampling exploits size and stiffness of system.
  - Can be transposed to τ-leaping algorithms (!)...

- **Treatment of negative species** with a tunable compromise efficiency-accuracy
  - An alternative to modified τ-leaping, which essentially recurs to SSA when in trouble

OUTLOOK

- R-leaping for stiff problems ?

E & Vanden-Eijnden,
J. Chem. Phys. 2005

Cao etal.,
J. Comp. Phys. 2005

- R-leaping of reaction-diffusion-convection problems