

# Conditioning by Permutation Monte Carlo for Continuous-Time Markov Chains

**Pierre L'Ecuyer**

Université de Montréal, Canada

joint work with

**Zdravko Botev**, New South Wales University, Australia

**Rohan Shah**, The University of Queensland, Brisbane, Australia

**Bruno Tuffin**, Inria–Rennes, France

Eighth International Workshop on Simulation, Vienna, September 2015

## Object of this talk

Think of a static system in which components have a random state, and there is a small probability  $u$  that their combined state is not good enough for the system to work properly.

Estimating  $u$  by standard Monte Carlo can be very inefficient.

In this talk we discuss a class of methods that transform the (simple) static model into a (more complicated) dynamic system, a continuous-time Markov chain, whose simulation can provide a much more efficient estimator of  $u$  when combined with conditional Monte Carlo.

## CTMC model

Some rare-event estimation problems can be recast as follows.

We have a continuous-time Markov chain (CTMC)  $\{\mathbf{X}(\gamma), \gamma \geq 0\}$  over state space  $\mathcal{X}$ , with  $\mathbf{X}(0) = \mathbf{x}_0 \in \mathcal{X}$ .

In state  $\mathbf{x} \in \mathcal{X}$ , jump rate is  $\lambda(\mathbf{x})$  and next state is  $\mathbf{x}'$  with (transition) probability  $p(\mathbf{x}' | \mathbf{x})$ . (Works also for a density  $p$ .)

## CTMC model

Some rare-event estimation problems can be recast as follows.

We have a continuous-time Markov chain (CTMC)  $\{\mathbf{X}(\gamma), \gamma \geq 0\}$  over state space  $\mathcal{X}$ , with  $\mathbf{X}(0) = \mathbf{x}_0 \in \mathcal{X}$ .

In state  $\mathbf{x} \in \mathcal{X}$ , jump rate is  $\lambda(\mathbf{x})$  and next state is  $\mathbf{x}'$  with (transition) probability  $p(\mathbf{x}' | \mathbf{x})$ . (Works also for a density  $p$ .)

Let  $T_0 = 0$  and  $T_1 < T_2 < \dots$  the jump times.

At  $T_j$ , chain jumps from state  $\mathbf{X}_{j-1} = \mathbf{X}(T_{j-1})$  to state  $\mathbf{X}_j = \mathbf{X}(T_j)$ . The associated discrete-time chain (DTMC) is  $\{\mathbf{X}_j, j \geq 0\}$ .

Let  $\Delta \subset \mathcal{X}$  (target) and  $C = \inf\{j \geq 0 : \mathbf{X}_j \in \Delta\}$  (a stopping time).

We want to estimate some function of  $T_C = \inf\{T_j : \mathbf{X}_j \in \Delta\}$ ; e.g.,  $u(\gamma) = \mathbb{P}[T_C > \gamma]$ .

Conditional on the DTMC trajectory  $\mathcal{H} = (\mathbf{X}_0, \mathbf{X}_1, \dots, \mathbf{X}_C)$ , the  $A_j = T_j - T_{j-1}$  are independent exponential with rates  $\Lambda_j = \lambda(\mathbf{X}_{j-1})$ .

## Monte Carlo (MC) for function of $T_C$

To estimate  $u = u(\gamma) = \mathbb{P}[T_C > \gamma]$ , generate  $n$  realizations  $l_1, \dots, l_n$  of  $l = \mathbb{I}[T_C > \gamma]$ , and compute estimator

$$\bar{l}_n = \frac{1}{n} \sum_{i=1}^n l_i.$$

When  $u$  is very small ( $\{T_C > \gamma\}$  is a **rare event**), direct MC fails because the rare event occurs too rarely.

## Monte Carlo (MC) for function of $T_C$

To estimate  $u = u(\gamma) = \mathbb{P}[T_C > \gamma]$ , generate  $n$  realizations  $I_1, \dots, I_n$  of  $I = \mathbb{I}[T_C > \gamma]$ , and compute estimator

$$\bar{I}_n = \frac{1}{n} \sum_{i=1}^n I_i.$$

When  $u$  is very small ( $\{T_C > \gamma\}$  is a **rare event**), direct MC fails because the rare event occurs too rarely.

### Relative error

$$\text{RE}[\bar{I}_n] \stackrel{\text{def}}{=} \frac{\sqrt{\text{MSE}[\bar{I}_n]}}{u} \stackrel{\text{here}}{=} \frac{\sqrt{1-u}}{\sqrt{nu}} \rightarrow \infty \quad \text{when } u \rightarrow 0.$$

For example, if  $u \approx 10^{-10}$ , we need  $n \approx 10^{12}$  to have  $\text{RE}[\bar{I}_n] \leq 10\%$ .

We would like **bounded RE** (or almost) when  $u \rightarrow 0$ .

## Conditioning on the DTMC

Alternative: replace  $I = \mathbb{I}[T_C > \gamma]$  by  $W = \mathbb{P}[T_C > \gamma \mid \mathcal{H}]$ .

Idea used for static network reliability estimation by Elperin, Gertsbach, Lomonosov [1974, 1991, 1992, etc.].

Conditional on  $\mathcal{H}$ ,  $T_C = A_1 + \dots + A_C$  is **hypoexponential**, with cdf and pdf:

$$F(\gamma \mid \mathcal{H}) = \mathbb{P}[T_C \leq \gamma \mid \mathcal{H}] = 1 - \mathbf{e}_1^t \exp(\mathbf{Q} \gamma) \mathbf{1}$$

and

$$f(\gamma \mid \mathcal{H}) = -\mathbf{e}_1^t \exp(\mathbf{Q} \gamma) \mathbf{Q} \mathbf{1},$$

where  $\mathbf{e}_1^t = (1, 0, \dots, 0)$ ,  $\mathbf{1} = (1, \dots, 1)^t$ , and

$$\mathbf{Q} = \begin{pmatrix} -\Lambda_1 & \Lambda_1 & 0 & \dots & 0 \\ 0 & -\Lambda_2 & \Lambda_2 & 0 & \vdots \\ 0 & 0 & \ddots & \ddots & 0 \\ \vdots & 0 & 0 & -\Lambda_{C-1} & \Lambda_{C-1} \\ 0 & \dots & 0 & 0 & -\Lambda_C \end{pmatrix}.$$

Higham (2009) has a stable and accurate algorithm for the **matrix exponential** that takes  $\mathcal{O}(C^3)$  time. Slow when  $C$  is large.



Higham (2009) has a stable and accurate algorithm for the **matrix exponential** that takes  $\mathcal{O}(C^3)$  time. Slow when  $C$  is large.

In some applications,  $\Lambda_j$  decreases strictly with  $j$ .

Then, explicit formulas ( $\mathcal{O}(c^2)$  time):

$$W = \mathbb{P}[T_C > \gamma \mid \mathcal{H}] = \sum_{j=1}^C e^{-\Lambda_j \gamma} p_j(\mathcal{H}) = 1 - \sum_{j=1}^C (1 - e^{-\Lambda_j \gamma}) p_j(\mathcal{H})$$

where

$$p_j(\mathcal{H}) = \prod_{k=1, k \neq j}^C \frac{\Lambda_k}{\Lambda_k - \Lambda_j}.$$

But does not work if some  $\Lambda_k - \Lambda_j = 0$  (or near).

## Permutation Monte Carlo estimator

To estimate  $u(\gamma) = \mathbb{P}[T_C > \gamma]$ , sample  $n$  independent realizations of  $\mathcal{H}$  and compute  $W = W(\mathcal{H}) = \mathbb{P}[T_C > \gamma \mid \mathcal{H}]$  for each, say  $W_1, \dots, W_n$ .

PMC estimator:

$$\bar{W}_n = \frac{1}{n} \sum_{i=1}^n W_i.$$

Since  $\mathbb{P}[0 < W < 1] = 1$ , one has  $\text{Var}[W] < \text{Var}[I]$  and

$$\text{Var}[\bar{W}_n] < \text{Var}[\bar{I}_n] \quad (\text{strictly}).$$

## Bounded relative error (BRE)

Suppose the distribution of  $T_C$  depends on a parameter  $\epsilon$  and  $u = u(\epsilon) = \mathbb{P}[T_C(\epsilon) > \gamma] \rightarrow 0$  when  $\epsilon \rightarrow 0$ . Then,

$$\text{RE}^2[\bar{I}_n] = \frac{\text{Var}[\bar{I}_n]}{u^2} = \frac{1 - u}{nu} \rightarrow \infty \quad \text{when } \epsilon \rightarrow 0.$$

## Bounded relative error (BRE)

Suppose the distribution of  $T_C$  depends on a parameter  $\epsilon$  and  $u = u(\epsilon) = \mathbb{P}[T_C(\epsilon) > \gamma] \rightarrow 0$  when  $\epsilon \rightarrow 0$ . Then,

$$\text{RE}^2[\bar{I}_n] = \frac{\text{Var}[\bar{I}_n]}{u^2} = \frac{1-u}{nu} \rightarrow \infty \quad \text{when } \epsilon \rightarrow 0.$$

However,  $\text{RE}^2[\bar{W}_n] = \mathbb{E}[W^2/u^2 - 1] < \infty$  iff  $\mathbb{E}[W^2/u^2] \leq K$ .

If the number of paths  $\mathcal{H}$  is finite, this happens iff when  $\epsilon \rightarrow 0$ ,  $\mathbb{P}[\mathcal{H}]W^2(\mathcal{H})/u^2 \leq K$  for all  $\mathcal{H}$ . Holds iff  $\mathbb{P}[\mathcal{H}] \not\rightarrow 0$  for all  $\mathcal{H}$ .

A sufficient condition: whenever  $p(\mathbf{x}' | \mathbf{x}) > 0$ ,  $p(\mathbf{x}' | \mathbf{x}) \not\rightarrow 0$  when  $\epsilon > 0$ .

More specific (weaker) conditions can be formulated on certain examples.

Our development adapts easily to the case where we want to estimate  $\mathbb{P}[T_C(\epsilon) \leq \gamma] \rightarrow 0$  instead of  $\mathbb{P}[T_C(\epsilon) > \gamma]$ .

## Simple example: Credit risk

An lending firm (or bank) has  $m$  obligors. With probability  $u_i$ , obligor  $i$  defaults ( $X_i = 1$ ) and the firm loses  $\ell_i$  from it. Total loss:

$$L = L(\mathbf{X}) = \sum_{i=1}^m X_i \ell_i.$$

We want to estimate  $u = \mathbb{P}[L > \ell^*]$ .

## Simple example: Credit risk

An lending firm (or bank) has  $m$  obligors. With probability  $u_i$ , obligor  $i$  defaults ( $X_i = 1$ ) and the firm loses  $\ell_i$  from it. Total loss:

$$L = L(\mathbf{X}) = \sum_{i=1}^m X_i \ell_i.$$

We want to estimate  $u = \mathbb{P}[L > \ell^*]$ .

Exact formula ( $2^m$  terms):

$$u = \sum_{\mathbf{x} \in \{0,1\}^m} \mathbb{I}[L(\mathbf{x}) > \ell^*] \prod_{i=1}^m u_i^{x_i} (1 - u_i)^{1-x_i}.$$

## Simple example: Credit risk

An lending firm (or bank) has  $m$  obligors. With probability  $u_i$ , obligor  $i$  defaults ( $X_i = 1$ ) and the firm loses  $\ell_i$  from it. Total loss:

$$L = L(\mathbf{X}) = \sum_{i=1}^m X_i \ell_i.$$

We want to estimate  $u = \mathbb{P}[L > \ell^*]$ .

Exact formula ( $2^m$  terms):

$$u = \sum_{\mathbf{x} \in \{0,1\}^m} \mathbb{I}[L(\mathbf{x}) > \ell^*] \prod_{i=1}^m u_i^{x_i} (1 - u_i)^{1-x_i}.$$

MC: generate  $\mathbf{X} = (X_1, \dots, X_m)$ , compute  $L$  and  $I = \mathbb{I}[L > \ell^*]$ . Repeat  $n$  times and take the average.

## From static model to CTMC

PMC: define  $Y_i$  exponential with rate  $\lambda_i = -\ln(1 - u_i)$ .

Then  $\mathbb{P}[Y_i \leq 1] = 1 - e^{-\lambda_i} = u_i$ .

Define a CTMC whose state is a binary vector  $\mathbf{X}(\gamma) = (X_1(\gamma), \dots, X_m(\gamma))$  where  $X_i(0) = 0$  and  $X_i(\gamma)$  jumps from 0 to 1 at time  $\gamma = Y_i$ .

We have  $\mathbb{P}[X_i(1) = 1] = u_i$  and  $\mathbb{P}[L(\mathbf{X}(1)) > \ell^*] = u$ .



## From static model to CTMC

**PMC:** define  $Y_i$  exponential with rate  $\lambda_i = -\ln(1 - u_i)$ .

Then  $\mathbb{P}[Y_i \leq 1] = 1 - e^{-\lambda_i} = u_i$ .

Define a CTMC whose state is a binary vector  $\mathbf{X}(\gamma) = (X_1(\gamma), \dots, X_m(\gamma))$  where  $X_i(0) = 0$  and  $X_i(\gamma)$  jumps from 0 to 1 at time  $\gamma = Y_i$ .

We have  $\mathbb{P}[X_i(1) = 1] = u_i$  and  $\mathbb{P}[L(\mathbf{X}(1)) > \ell^*] = u$ .

Let  $\pi$  the permutation s.t.  $Y_{\pi(1)} < \dots < Y_{\pi(m)}$  (ordered jumps).

Conditional on  $\pi$ , we can forget the  $Y_i$ 's, switch the  $X_i$ 's one by one until  $L(\mathbf{X}) > \ell^*$ , say at step  $C = \min\{j : L(\mathbf{X}_j) > \ell^*\}$ .

The CTMC has initial jump rate  $\Lambda_1 = \lambda_1 + \dots + \lambda_m$ .

At the  $j$ th jump of the DTMC,  $X_{\pi(j)}$  switches from 0 to 1 and the jump rate decreases to  $\Lambda_{j+1} = \Lambda_j - \lambda_{\pi(j)}$ . Thus,  $\Lambda_j$  is strictly decreasing in  $j$  and  $C$  cannot exceed  $m$ . **Easy to apply PMC.**

## A CTMC that removes the defaults

Instead of adding the defaults one by one, we can construct a CTMC that starts with all obligors in default, and remove the defaults one by one until  $L(\mathbf{X}) \leq \ell^*$ , say at step  $C'$ .

The “anti-default” that switches  $X_i$  from 1 to 0 occurs at time  $R_i$ , taken as exponential with rate  $\mu_i = -\ln(1 - e^{-\lambda_i})$ , so  $\mathbb{P}[R_i \leq 1] = \mathbb{P}[Y_i > 1]$ .

We have a similar PMC estimator of  $u$ , with  $C'$ ,  $\Lambda'_j$ , ..., defined from the  $\mu_i$ :

$$W' = \mathbb{P}[A'_1 + \dots + A'_{C'} > 1 \mid \pi'].$$

In this example, we expect  $C \ll C'$ , i.e., we should need much less than half the obligors to default, so the first approach should be faster.

But the second approach may win in terms of variance.

## BRE, functional estimation, etc.

Suppose  $u_j = u_j(\epsilon) \rightarrow 0$  when  $\epsilon \rightarrow 0$ .

If  $u_i(\epsilon)/u_j(\epsilon) \geq \delta > 0$  for all  $i, j$ , then the PMC estimator has bounded relative error (sufficient condition for BRE).

## BRE, functional estimation, etc.

Suppose  $u_i = u_i(\epsilon) \rightarrow 0$  when  $\epsilon \rightarrow 0$ .

If  $u_i(\epsilon)/u_j(\epsilon) \geq \delta > 0$  for all  $i, j$ , then the PMC estimator has bounded relative error (**sufficient condition for BRE**).

Moreover, if  $u_i(\epsilon) = \kappa\epsilon$  for all  $i$ , then the law of the DTMC (or  $\pi$ ) does not depend on  $\epsilon$ , so we can estimate  $u(\epsilon)$  as a function of  $\epsilon$  (the **entire function**), and even its **derivative w.r.t.  $\epsilon$** , from a single sample of  $n$  realizations of  $\pi$ .

## Multicomponent static system

A system has  $m$  components, in state 0 (failed) or 1 (operating).

System state:  $\mathbf{X} = (X_1, \dots, X_m)^t$ .

Structure function:  $\Phi : \{0, 1\}^m \rightarrow \{0, 1\}$ , assumed monotone.

System is operational iff  $\Phi(\mathbf{X}) = 1$ .

Unreliability:  $u = \mathbb{P}[\Phi(\mathbf{X}) = 0]$ .

## Multicomponent static system

A system has  $m$  components, in state 0 (failed) or 1 (operating).

System state:  $\mathbf{X} = (X_1, \dots, X_m)^t$ .

Structure function:  $\Phi : \{0, 1\}^m \rightarrow \{0, 1\}$ , assumed monotone.

System is operational iff  $\Phi(\mathbf{X}) = 1$ .

Unreliability:  $u = \mathbb{P}[\Phi(\mathbf{X}) = 0]$ .

**CTMC model:** Suppose all links are initially operational. For certain subsets  $\mathbf{s} \subseteq \{1, \dots, m\}$ , a shock that takes down all components in  $\mathbf{s}$  occurs at time  $Y_j$ , an exponential with rate  $\lambda_j = \lambda_{\mathbf{s}(j)}$ .

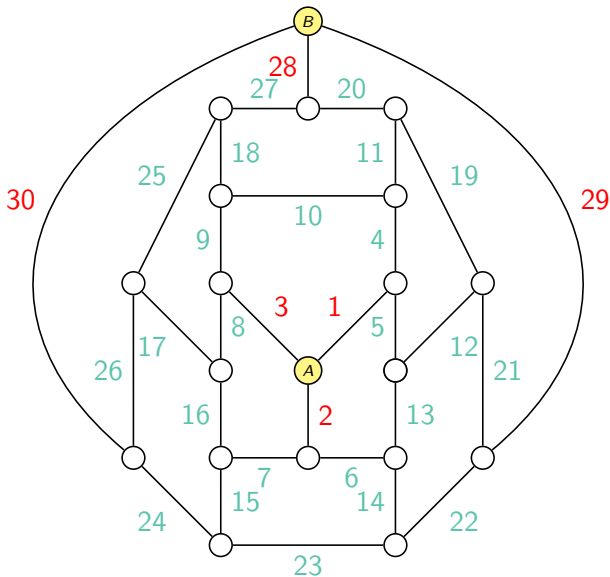
At time  $Y_j$ ,  $X_i(\gamma)$  is set to 0 for all  $i \in \mathbf{s}(j)$ .

We have  $u = \mathbb{P}[\Phi(\mathbf{X}(1)) = 0]$ .

**PMC:** Generate only the order in which the shocks occur, and compute  $W$ .

**Scanning the shocks in reverse order:** We can also replace shocks by “anti-shocks”: Start by assuming that all the shocks have occurred and remove them one by one.

## Example: A dodecahedron network



# Dodecahedron graph, shocks on nodes and on links

All shocks at rate  $\lambda_j$  except on  $\mathcal{V}_0 = \{1, 20\}$ ;  $n = 10^6$ .

algorithm	$\bar{W}_n$	$S_n^2/\bar{W}_n^2$	$RE[\bar{W}_n]$	$\bar{C}$	$T(\text{sec})$	WNRV
$\lambda_j = 10^{-3}$						
PMC	1.62e-8	993	0.032	12.7	35	0.035
PMC-anti	1.60e-8	1004	0.032	36.3	17	0.018
turnip	1.63e-8	894	0.030	10.7	● 72	0.064
turnip-anti	1.58e-8	296	0.017	35.8	○ 45	0.013
$\lambda_j = 10^{-7}$						
PMC	1.65e-20	1047	0.032	12.7	32	0.034
PMC-anti	1.66e-20	1044	0.032	36.3	18	0.019
turnip-anti	1.58e-20	311	0.018	35.8	○ 44	0.014



### Example: Max flow in network with random capacities

A network has links with random capacities, distributed over finite sets.

We want to estimate the probability  $u$  that the max flow between two given nodes is under a given target threshold.

In the CTMC model, the capacities start at their minimum, and at the jump times  $Y_j$ , the capacity of one or more link(s) jumps up.

Or it can be the other way around: capacities start at max and go down.

PMC can be very effective when  $u$  is very small and the network is not too large.

### Example: Probability of overflow in a communication network where links have fixed capacities and demand is random.

## Conclusion

The CTMC-PMC approach can be very efficient in situations where the number of steps in the CTMC remains relatively small, so the important permutations are not too rare.

The method is **not always** effective.

When  $C$  can be very large, important realizations of  $\mathcal{H}$  may be too rare, and other approaches could be needed. Splitting and generalized splitting (GS) often work well in those situations. See our papers (on our web pages) for further details.