



# Transient Analysis of Markov Chains by Controlled Probability Mass Selection

José Manuel Martínez, Reinaldo Vallejos, Gerardo Rubino

## ► To cite this version:

José Manuel Martínez, Reinaldo Vallejos, Gerardo Rubino. Transient Analysis of Markov Chains by Controlled Probability Mass Selection. AMS Spring Eastern Sectional Meeting 2023, AMS, Apr 2023, virtual, United States. pp.1-26. hal-04381190

**HAL Id: hal-04381190**

**<https://inria.hal.science/hal-04381190>**

Submitted on 10 Jan 2024

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Public Domain

# Transient Analysis of Markov Chains by Controlled Probability Mass Selection

José M. Martínez, Reinaldo A. Vallejos, Gerardo Rubino

José Manuel Martínez

Department of Electronic Engineering  
Universidad Técnica Federico Santa María. Valparaíso, Chile

AMS Spring Eastern Sectional Meeting 2023  
(Virtual)

April 1-2, 2023

- ➊ Motivation
- ➋ Background: CTMC Transient Analysis by Uniformization
- ➌ Error Description/Analysis
- ➍ Our Proposal to Improve the Uniformization Method
- ➎ Conclusions

- **Continuous-time Markov chains** (CTMCs) are widely used to model real-world systems and processes in many areas such as (amongst many others)
  - engineering and environmental processes
  - computer and communication networks
  - production and manufacturing systems
  - healthcare systems
  - epidemic diseases
  - biochemically reacting systems
- Often the interest is in **transient (time dependent) system behavior** or performance properties, respectively, and the analysis requires the determination of transient probability distributions

**Uniformization**, also known as **randomization** or **Jensen's method**, provides a numerically stable computational scheme to approximate the **transient probability distributions** of CTMCs with uniformly bounded transition rates

- Reliability/Availability analysis usually involves systems that were design for high availability: system is working [properly] most of the time.
- The state space could be seen as a partition with "working" states and "failure" states
- In these systems, most of the probability mass is [generally] concentrated in a small subset of states
- The same [usually] occurs in queuing systems

Can we take advantage of this behavior, in order to discard some probability mass during the computation of transient measures, and simultaneously control the introduced error?

## Continuous Time Markov Chain (CTMC)

Let  $\mathcal{X} = \{X(t), t \geq 0\}$  be a **CTMC** with

- $\mathcal{S}$ : finite state-space
- $\mathbf{Q}$ : infinitesimal generator matrix
- $\pi_i(t) = P[X(t) = i]$
- $\vec{\pi}(t) = [\pi_1(t), \dots, \pi_{|\mathcal{S}|}(t)]$ : transient probability vector

## System Dynamics:

$$\frac{d}{dt} \vec{\pi}(t) = \vec{\pi}(t) \mathbf{Q},$$

$\vec{\pi}(0)$  : initial condition

## System Dynamics: Differential Equations

$$\frac{d}{dt}\vec{\pi}(t) = \vec{\pi}(t)\mathbf{Q},$$

$\vec{\pi}(0)$  : initial condition

## System Dynamics: Solution

- Direct:

$$\vec{\pi}(t) = \vec{\pi}(0)e^{\mathbf{Q}t}$$

- Taylor Series Expansion:

$$\vec{\pi}(t) = \vec{\pi}(0) \sum_{n=0}^{\infty} \frac{(\mathbf{Q}t)^n}{n!}$$

$\mathbf{Q}$  contains negative and unbounded entries, which implies that the evaluation of  $e^{\mathbf{Q}t}$  is numerically unstable.

## MC Transient Analysis: Uniformization

### Direct Solution $\rightarrow$ Taylor Series Expansion + Stability

$$\vec{\pi}(t) = \vec{\pi}(0) e^{\mathbf{Q}t}$$

**Idea:** define the stochastic matrix  $\mathbf{P} = \mathbf{Q}/\Lambda + \mathbf{I}$ , with  $\Lambda \geq \max_i(|Q_{i,i}|)$ , then

### Uniformization (a.k.a. Randomization or Jensen's Method)

$$\vec{\pi}(t) = \sum_{n=0}^{\infty} e^{-\Lambda t} \frac{(\Lambda t)^n}{n!} \vec{\pi}(0) \mathbf{P}^n$$

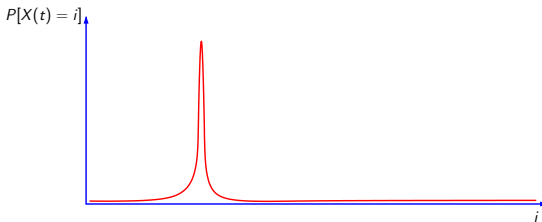
which can be computed efficiently by,

$$\vec{\pi}(t) = \sum_{n=0}^{\infty} e^{-\Lambda t} \frac{(\Lambda t)^n}{n!} \vec{\nu}(n), \quad \text{where} \quad \vec{\nu}(n) = \vec{\nu}(n-1) \mathbf{P}, \\ \vec{\nu}(0) = \vec{\pi}(0)$$

$\vec{\nu}(n)$  is the transient probability vector of the *uniformized DTMC*, characterized by  $\mathbf{P}$



In availability/reliability and queueing models is usual to have the probability mass concentrated in a small subset of the state space.



### Improving Uniformization: Strategy

- discarding probability mass that is considered negligible for the overall computation process ( $\vec{\nu}(n) = \vec{\nu}(n-1) \mathbf{P}$ ).
- using a data structure to efficiently manage those relevant entries of  $\vec{\nu}(n)$ .

Uniformization:

$$\vec{\pi}(t) = \sum_{n=0}^{\infty} e^{-\Lambda t} \frac{(\Lambda t)^n}{n!} \vec{\nu}(n)$$

$$\vec{\nu}(n) = \vec{\nu}(n-1) \mathbf{P},$$

$$\vec{\nu}(0) = \vec{\pi}(0)$$

1) **Truncation** on  $n = N$  (uniformization steps): well known technique

$$\vec{\pi}(t) = \sum_{n=0}^N e^{-\Lambda t} \frac{(\Lambda t)^n}{n!} \vec{\nu}(n) + \vec{\epsilon}_t(N)$$

2) **Selection** of probability mass of  $\vec{\nu}(n) \leftarrow$  **contribution**

$$\vec{\pi}(t) = \sum_{n=0}^N e^{-\Lambda t} \frac{(\Lambda t)^n}{n!} \hat{\vec{\nu}}(n) + \vec{\epsilon}_t(N) + \vec{\epsilon}_d(N)$$

$\vec{\epsilon}_d(N)$ : vector containing to the total discarded probability mass

## Approximation: Actual computation

$$\hat{\vec{\pi}}(t) = \sum_{n=0}^N e^{-\Lambda t} \frac{(\Lambda t)^n}{n!} \hat{\vec{\nu}}(n)$$

## Error Analysis

$$\begin{aligned} \|\vec{\pi}(t) - \hat{\vec{\pi}}(t)\|_1 &\leq \|\vec{\epsilon}_t(N)\|_1 + \|\vec{\epsilon}_d(N)\|_1 \\ &\leq \varepsilon_t + \varepsilon_d \end{aligned}$$

- Error due to truncation of infinite summation

$$\|\vec{\epsilon}_t(N)\|_1 \leq \varepsilon_t$$

There are several techniques to find out  $N$  and to calculate (accurately)  $e^{-\Lambda t} \frac{(\Lambda t)^n}{n!}$ , by specifying a value of  $\varepsilon_t$ , beforehand.

- Error due to selection of some entries from the probability vector

$$\|\vec{\epsilon}_d(N)\|_1 \leq \varepsilon_d$$

**can be specified beforehand?**

## Effect of discarding probability mass on a DTMC

### Analyzing $\vec{\nu}(n) = \vec{\nu}(n-1) \mathbf{P}$

- Let  $\vec{\delta}(n)$  be the discarding probability vector of the DTMC at step  $n$ , with a total probability mass discarded of  $\delta_n$  at that step, i.e.,

$$\|\vec{\delta}(n)\|_1 = \delta_n$$

- Let  $\hat{\vec{\nu}}(n)$  be the approximate vector of  $\vec{\nu}(n)$ , after discarding the mass  $\vec{\delta}(n)$ , i.e.,

$$\vec{\nu}(n) = \hat{\vec{\nu}}(n) + \vec{\delta}(n)$$

- Therefore, after  $m$  iteration steps, we have:

$$\vec{\nu}(n+m) = \vec{\nu}(n) \mathbf{P}^m = \hat{\vec{\nu}}(n) \mathbf{P}^m + \vec{\delta}(n) \mathbf{P}^m$$

- Let  $\vec{\epsilon}_d(n, m) = \vec{\delta}(n) \mathbf{P}^m$  be the error after  $m$  iteration steps, introduced by discarding  $\vec{\delta}(n)$  at step  $n$ . Consequently,

$$\|\vec{\epsilon}_d(n, m)\|_1 = \|\vec{\delta}(n) \mathbf{P}^m\|_1 = \|\vec{\delta}(n)\|_1 = \delta_n$$

The error (norm-1) is the total probability mass discarded at step  $n$  and does not change with the evolution of the DTMC

Selection of probability mass

$$\vec{\pi}(t) = \sum_{n=0}^N e^{-\Lambda t} \frac{(\Lambda t)^n}{n!} \hat{\vec{\nu}}(n) + \vec{\epsilon}_t(N) + \vec{\epsilon}_d(N)$$

## Selection: Strategy

Let  $\delta_n$  be the discarded probability mass of  $\hat{\vec{\nu}}(n)$  at step  $n$ . Then, we have to comply with:

$$\sum_{n=0}^N \delta_n \leq \|\vec{\epsilon}_d(N)\|_1$$

Discarding  $\delta_n$  from  $\hat{\vec{\nu}}(n)$ , is equivalent to **select** (preserve) a probability mass of:

$$\omega_n = \|\hat{\vec{\nu}}(n)\|_1 - \delta_n$$

This is the point of view that we will consider hereafter

### Question

How do we select a probability mass of  $\omega_n$  at each uniformization step?

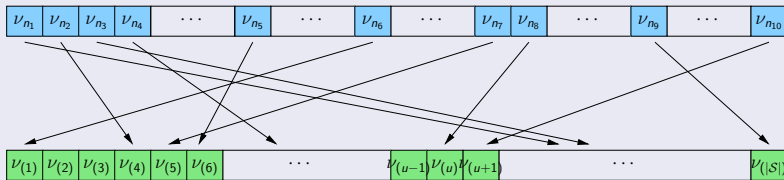
### Question

How do we select a probability mass of  $\omega_n$  at each uniformization step?

### Answer

Selecting, in "order", from largest down to the smallest vector component

## Sorting and Selection of entries of $\vec{\nu}(n)$



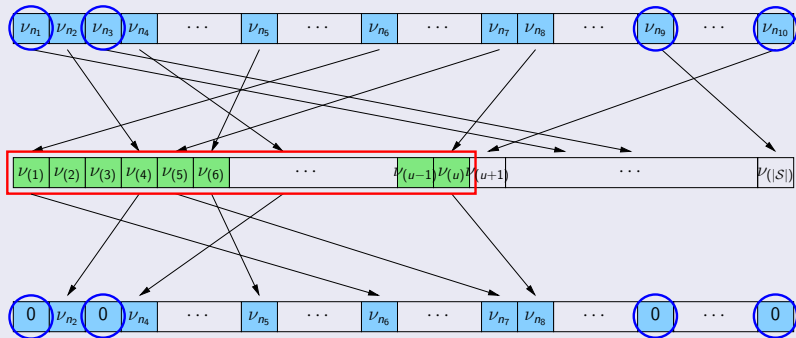
## Conditions

- Sorting:  $\nu_{(1)} \geq \nu_{(2)} \geq \nu_{(3)} \geq \dots \geq \nu_{(u-1)} \geq \nu_{(u)} \geq \dots \geq \nu_{(|S|)}$



# MC Transient Analysis: Selection Approach

## Sorting and Selection of entries of $\vec{\nu}(n)$



## Conditions

- Sorting:  $\nu_{(1)} \geq \nu_{(2)} \geq \nu_{(3)} \geq \dots \geq \nu_{(u-1)} \geq \nu_{(u)} \geq \dots \geq \nu_{(|S|)}$
- Selection:  $\min_u \left\{ u \mid \sum_{i=1}^u \nu_{(i)} \geq \omega_n \right\}$

## Comments

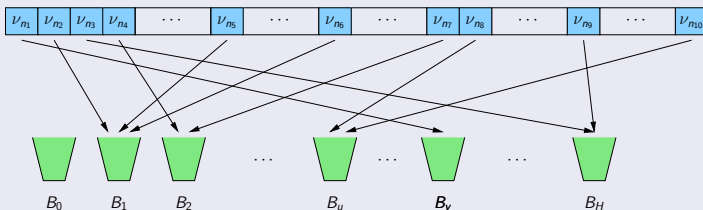
- Sorting all elements of an array of dimension  $|\mathcal{S}|$  is a computationally expensive task ( $|\mathcal{S}| \log_2 |\mathcal{S}|$ , in average)
- The high computational cost of sorting the probability vector (in each step), have discouraged researchers to consider this approach to deal with negligible probabilities
- We propose an efficient method for "sorting" the probability vector

## Other approaches

An alternative approach, that has been used recently, consists in just discarding those probabilities that are below a certain threshold, fixed by the user. While it speeds up computation of uniformization, there is no formal way of defining a threshold and the error is not predictable

## Probability Mass Selection: Strategy (\*)

- ❶ A number  $x$  can be represented in any base as:  $x = \pm b^{c_x \cdot m_x}$ , where  $b$  is the base,  $c_x$  is the characteristic, and  $m_x$  is the mantissa
- ❷ Define a bucket-selection algorithm where:
  - $g(x) = |c_x|$ , (for  $x > 0$ ).
  - $B_k$ : bucket (set) to store those entries  $\nu_i$ , where  $g(\nu_i) = k$ .

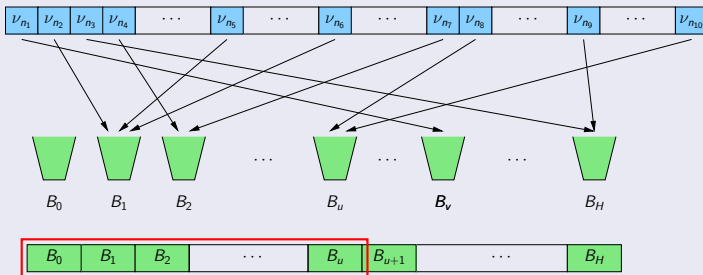


There exist a strict order relation among the buckets  $B_j, j \in \{0, \dots, H\}$ :

All elements of  $B_m$  are smaller than all elements of  $B_n$ , for  $m > n$

# MC Transient Analysis: Selection Approach

Selection by number representation of entries of  $\vec{\nu}(n)$

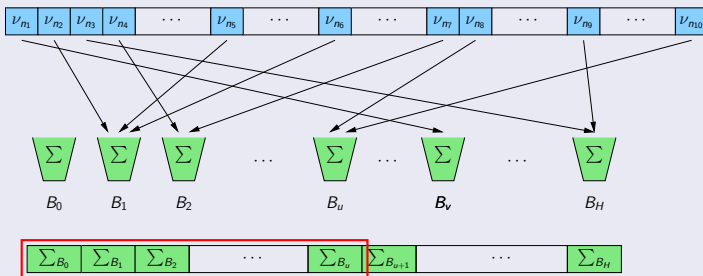


Comments:

- Selection:  $\min_u \left\{ u \mid \sum_{i=0}^u (\Sigma_{B_i}) \geq \omega_n \right\}$ , where  $\Sigma_{B_i} = \sum_{j: \nu_j \in B_i} \nu_j$
- The optimum value to be selected ( $\nu_{(u)}$ ) is in the bucket  $B_u$

# MC Transient Analysis: Selection Approach

## Selection by number representation of entries of $\vec{\nu}(n)$



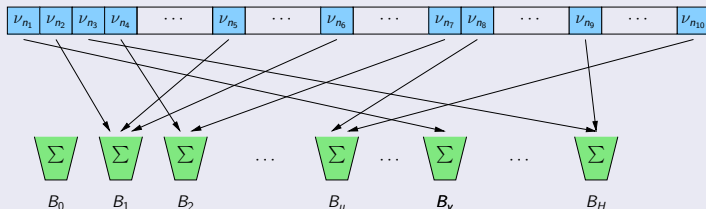
- Once we determine  $B_u$ , or equivalently,  $u$ , the selection of the components that satisfy the condition given by  $\omega_n$ , can be done by picking up those entries that comply with:

$$g(\nu_i) \leq u$$

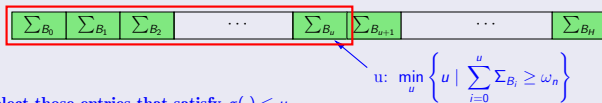
# MC Transient Analysis: Selection Approach

## Full scheme of selection

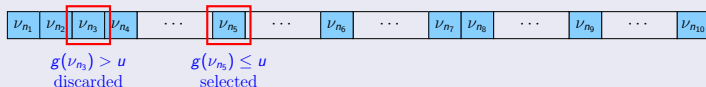
1) Select and sum in corresponding bucket the entries of  $\vec{\nu}(n)$



2) Determine  $u$  ( $B_u$ )



3) Select those entries that satisfy  $g(\cdot) \leq u$



## Uniformization by Probability Mass Selection

### 1 Uniformization (approximate)

$$\vec{\pi}(t) = \sum_{n=0}^N e^{-\Lambda t} \frac{(\Lambda t)^n}{n!} \hat{\vec{v}}(n) + \vec{\epsilon}_t(N) + \vec{\epsilon}_d(N)$$

### 2 Error due to the truncation of infinite summation

$$\|\vec{\epsilon}_t(N)\|_t \leq \varepsilon_t: \quad e^{-\lambda t} \frac{(\lambda t)^N}{N!}, N \leftarrow \varepsilon_t \text{ (known technique)}$$

### 3 Error due to the discarding procedure of probability mass

$$\|\vec{\epsilon}_d(N)\|_1 \leq \varepsilon_d: \quad \text{choose } \omega_n, \text{ such that } \sum_{n=0}^N \|\hat{\vec{v}}(n)\|_1 - \omega_n \leq \varepsilon_d$$

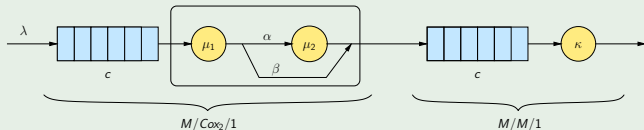
### 4 Evaluation of $\hat{\vec{v}}(n) = \hat{\vec{v}}(n-1)\mathbf{P}$

$$\hat{\vec{v}}(n) = \sum_{i: g(\hat{v}_i(n-1)) \leq u} \hat{v}_i(n-1) \vec{P}_i$$

$$g(b^{c_x \cdot m_x}) = |c_x|, \quad u: \min_u \left\{ u \mid \sum_{i=0}^u \Sigma_{B_i} \geq \omega_n \right\}, \quad \Sigma_{B_i} = \sum_{j: g(\hat{v}_j(n-1))=i} \hat{v}_j(n-1)$$

# MC Transient Analysis: Example

## Two Tandem Queues



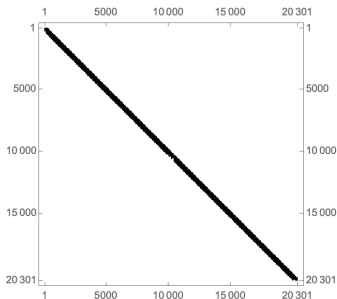
source: H. Hermanns, J. Meyer-Kayser and M. Siegle. "Multi-terminal binary decision diagrams to represent and analyse continuous-time Markov chains". In Proc. 3rd Int. Workshop on the Num. Sol. of Markov Chains, pp. 188-207, 1999

### Parameters:

- $c = 100$
- $\lambda = 4c$
- $\mu_1 = 2$
- $\mu_2 = 2$
- $\alpha = 0.1$
- $\beta = 0.9$
- $\kappa = 4$

starting condition: empty system (state 0)

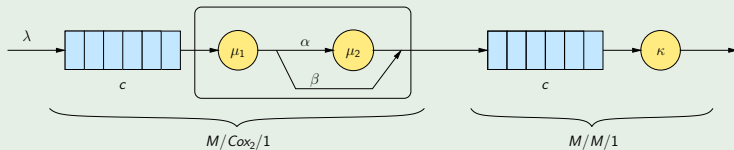
### Matrix Q





# MC Transient Analysis: Example

## Two Tandem Queues



State probabilities after  $t = 1$



*probability vs. state number*

Number of states (probability mass)



*number of states vs. uniformization step*

- This technique allows us to take advantage of the concentration of probability mass in few states
- We provide a bound (beforehand) on the numerical error, and the procedure guarantees that its answer complies with that specification
- If the probability mass is not concentrated, our method could perform worse than the standard uniformization (i.e., without discarding probabilities)
- A hybrid approach can be used (standard uniformization + our approach) in order to avoid the under-performance of the discarding procedure

# MC Transient Analysis: Note on Implementation

Saving in the evaluation of  $\vec{\nu}(n)$

$$\vec{\nu}(n) = \vec{\nu}(n-1) \mathbf{P}, \quad \vec{\nu}(0) = \vec{\pi}(0)$$

Consider  $\mathbf{P}$  in terms of row-vectors:

$$\mathbf{P} = \begin{bmatrix} \vec{P}_1 \\ \vec{P}_2 \\ \vdots \\ \vec{P}_{|S|} \end{bmatrix}, \quad \vec{P}_i = [P_{i,1}, P_{i,2}, \dots, P_{i,|S|}]$$

Evaluation of  $\vec{\nu}(n)$  can be written:

$$\vec{\nu}(n) = \sum_{i \in S} \nu_i(n-1) \vec{P}_i$$

