Uncertainty Quantification Using Cloud Computing for Monte Carlo Parallelization

Americo Cunha Jr[1], Rubens Sampaio[1],
Rafael Nasser[2], Hélio Lopes[2], and Karin Breitman[2]

[1]Department of Mechanical Engineering
[2]Department of Informatics
Pontifícia Universidade Católica do Rio de Janeiro

22th International Congress of Mechanical Engineering
November 3-7, 2013
Ribeirão Preto, SP, Brazil

Outline

Modeling and uncertainties

Computer models for prediction and decision making:

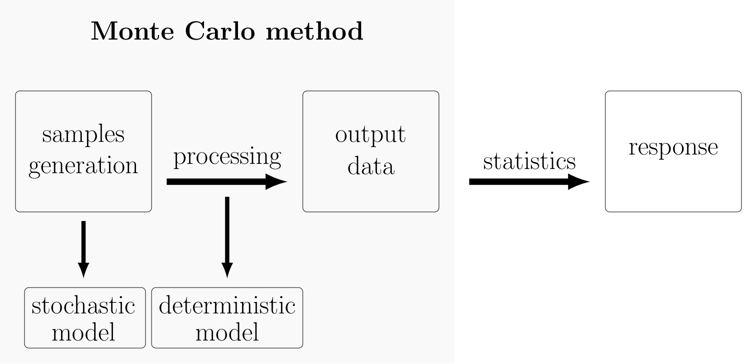
- engineering
- economics
- actuarial sciences
- etc

These models are subjected uncertainties due to:

- variability in the model parameters
- inaccuracies in model conception

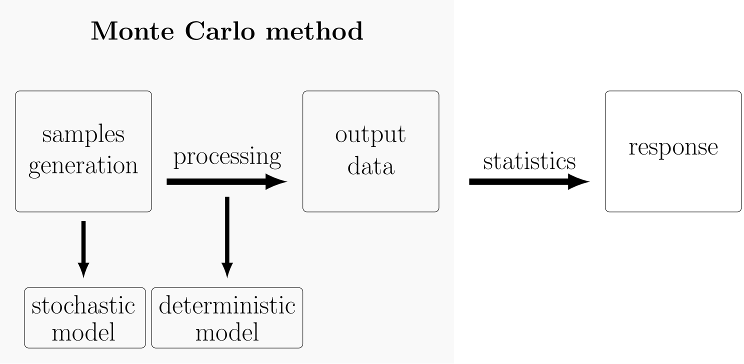
Propagation of uncertainties: Monte Carlo method

The most used technique to UQ is the Monte Carlo method.



Propagation of uncertainties: Monte Carlo method

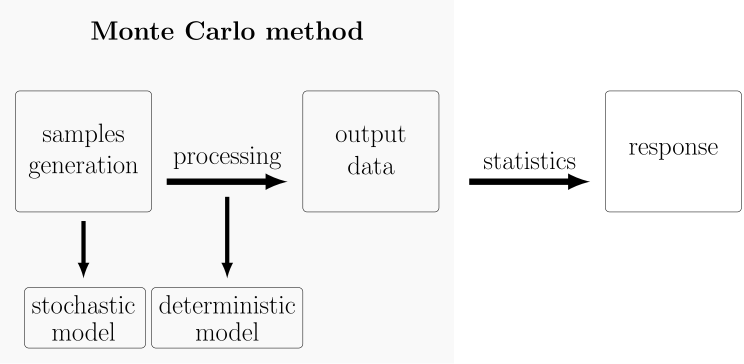
The most used technique to UQ is the Monte Carlo method.



- ☺ easy to implement

Propagation of uncertainties: Monte Carlo method

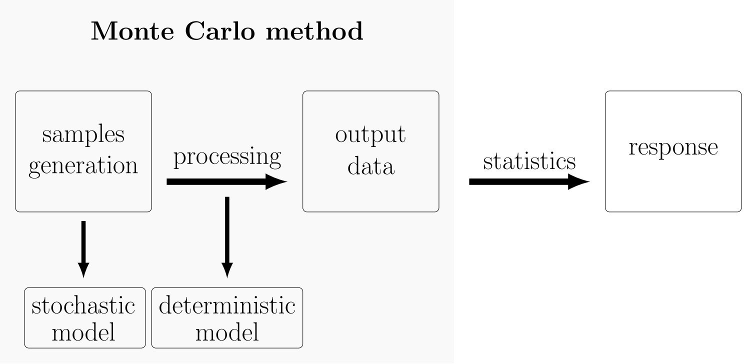
The most used technique to UQ is the Monte Carlo method.



- ☺ easy to implement
- ☺ good statistical results

Propagation of uncertainties: Monte Carlo method

The most used technique to UQ is the Monte Carlo method.



- ☺ easy to implement
- ☺ good statistical results
- ☹ extremely high computational cost

Research objectives

This work intends to:

- Present a cloud computing setting for MC parallelization
- Illustrate its usage in a stochastic structural dynamics problem
- Discuss the benefits of this parallelization approach

PUC RIO

A cloud computing setting for MC simulation

McCloud: a cloud computing setting for MC parallelization

- inspired in the MapReduce paradigm
- runs in Microsoft Windows Azure platform
- Divide MC simulation into three parts:
    1. split
    2. process
    3. merge

Parallelization strategy

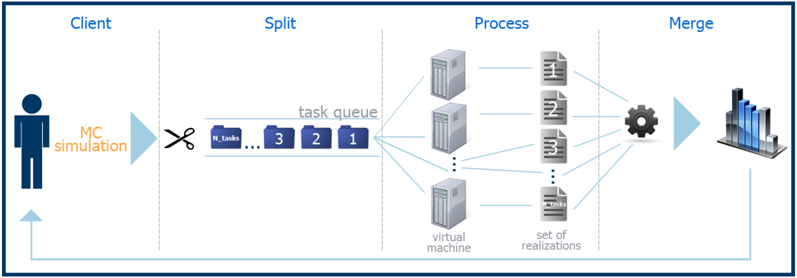


Figure: Overview of the cloud parallelization strategy for Monte Carlo.

Physical system: fixed-mass-spring bar

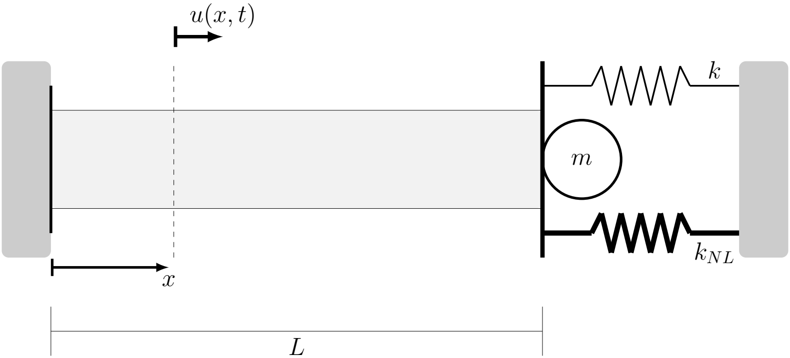


Figure: Sketch of a bar fixed at one and attached to two springs and a lumped mass on the other extreme, i.e. a fixed-mass-spring bar.

Mathematical formulation for the nonlinear dynamics

Find a "suitable" displacement field $u(x,t)$ that satisfies

$$\rho A \frac{\partial^2 u}{\partial t^2} + c\,\frac{\partial u}{\partial t} - \frac{\partial}{\partial x}\left(EA\,\frac{\partial u}{\partial x}\right) + \left(ku + k_{NL} u^3 + m\,\frac{\partial^2 u}{\partial t^2}\right)\delta(x-L) = f(x,t),$$

the boundary conditions

$$u(0,t) = 0, \quad \text{and} \quad EA\,\frac{\partial u}{\partial x}(L,t) = 0,$$

and the initial conditions

$$u(x,0) = u_0(x), \quad \text{and} \quad \frac{\partial u}{\partial t}(x,0) = v_0(x).$$

Random external force

The external force is modeled as the random field such that

$$F(\omega, x, t) = \sigma \sin\left(\lambda_1 \frac{x}{L}\right) N(\omega, t),$$

where the $N(\omega, t)$ is a normalized Gaussian white noise.

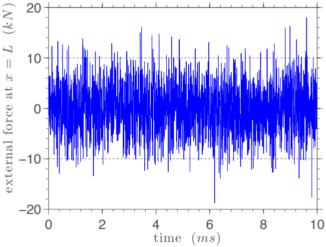


Figure: This figure illustrates a Gaussian white noise realization.

Random elastic modulus

The elastic modulus is modeled as a gamma distributited random variable, which the distribution was chosen by the maximum entropy principle.

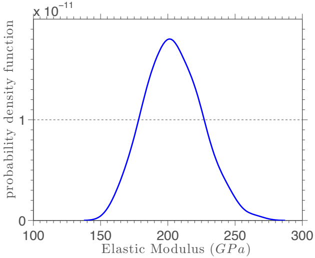


Figure: This figure illustrates the elastic modulus PDF.

Model equation discretization

For discretization, the Galerkin method is employed, which results is a nonlinear initial value problem of the form

$$[M]\,\ddot{\mathbf{u}}(t) + [C]\,\dot{\mathbf{u}}(t) + [K]\,\mathbf{u}(t) = \mathbf{f}(t) + \mathbf{f}_{NL}\left(\mathbf{u}(t)\right),$$

$$\mathbf{u}(0) = \mathbf{u}_0 \quad \text{and} \quad \dot{\mathbf{u}}(0) = \mathbf{v}_0,$$

which is integrated using Newmark method.

Probability density function estimation

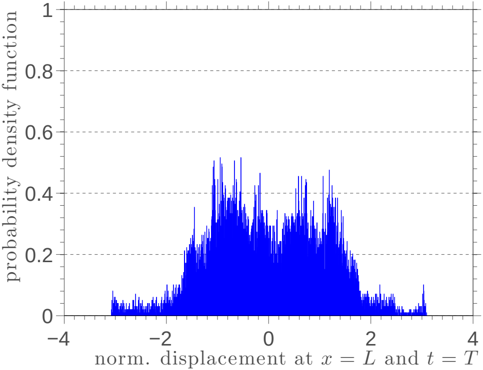


Figure: This figure illustrates an estimation a (normalized) displacement PDF, with 16 384 realizations.

Probability density function estimation

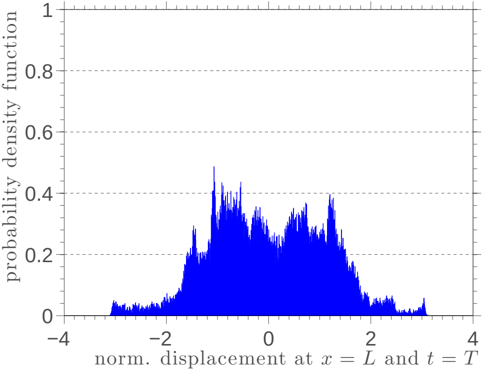


Figure: This figure illustrates an estimation a (normalized) displacement PDF, with 65 536 realizations.

Probability density function estimation

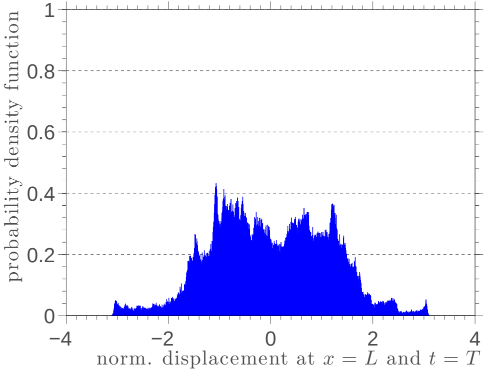


Figure: This figure illustrates an estimation a (normalized) displacement PDF, with 262 144 realizations.

Probability density function estimation

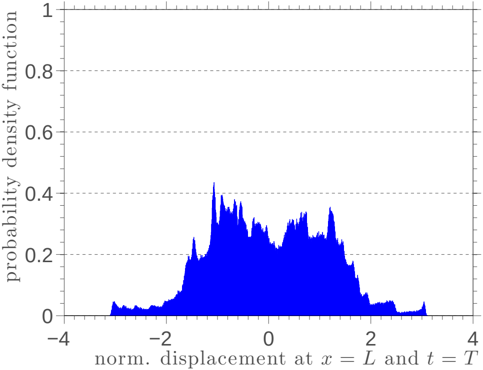
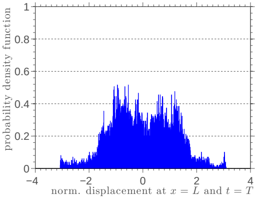
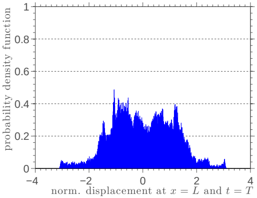


Figure: This figure illustrates an estimation a (normalized) displacement PDF, with 1 048 576 realizations.

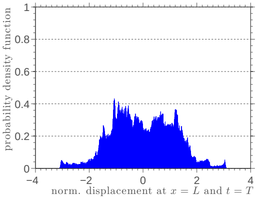
Probability density function estimation



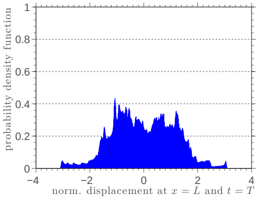
(a) 16 384 realizations



(b) 65 536 realizations



(c) 262 144 realizations



(d) 1 048 576 realizations

Residue of the diplacement PDF estimation

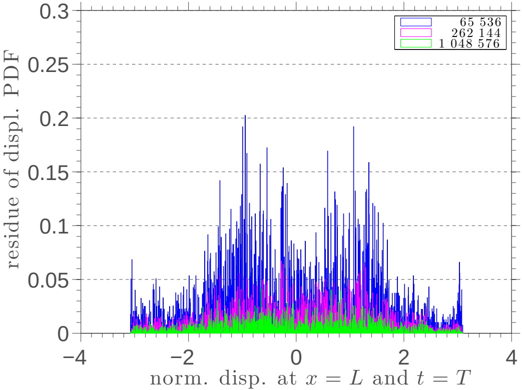


Figure: This figure illustrates the residue of the (normalized) displacement PDF.

Mean and standard deviation estimation

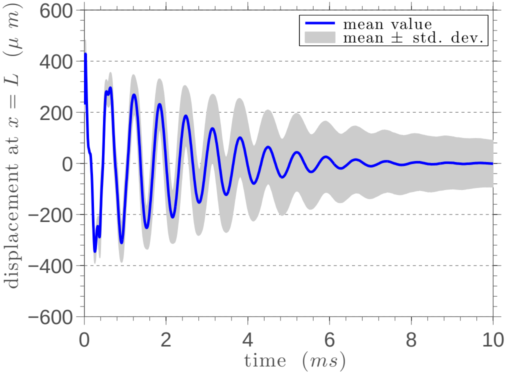


Figure: This figure illustrates the mean value (blue line) and an one standard deviation interval (grey shadow) for a (normalized) displacement, with 16 384 realizations.

Mean and standard deviation estimation

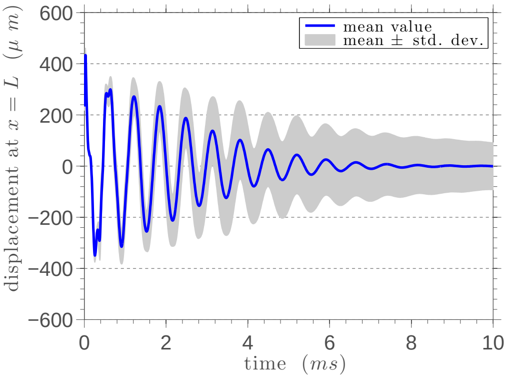


Figure: This figure illustrates the mean value (blue line) and an one standard deviation interval (grey shadow) for a (normalized) displacement, with 65 536 realizations.

Mean and standard deviation estimation

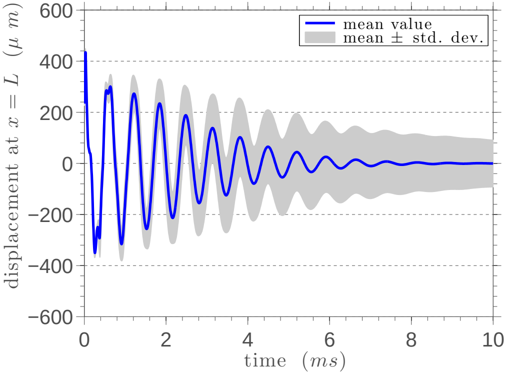


Figure: This figure illustrates the mean value (blue line) and an one standard deviation interval (grey shadow) for a (normalized) displacement, with 262 144 realizations.

Mean and standard deviation estimation

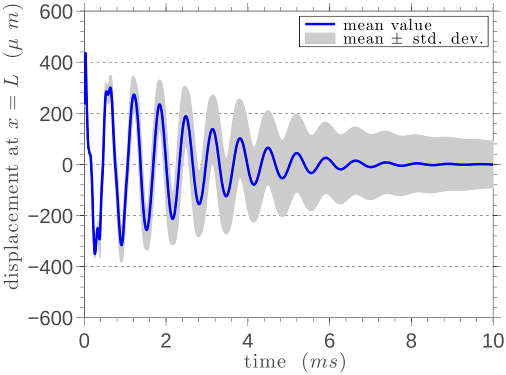
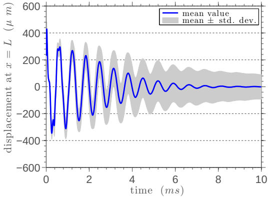
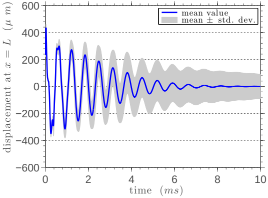


Figure: This figure illustrates the mean value (blue line) and an one standard deviation interval (grey shadow) for a (normalized) displacement, with 1 048 576 realizations.

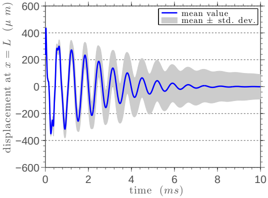
Mean and standard deviation estimation



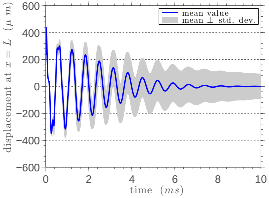
(a) 16 384 realizations



(b) 65 536 realizations



(c) 262 144 realizations



(d) 1 048 576 realizations

Residue of the diplacement mean estimation

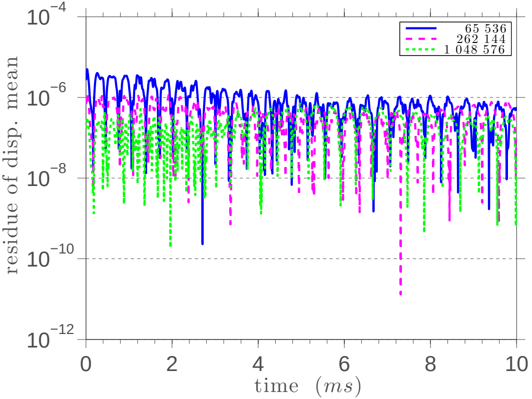


Figure: This figure illustrates the evolution of residue of the (normalized) displacement mean.

Costs analysis of the computational time

Table: Computational time spent by each MC simulation and the speed-up factors compared to a serial simulation (using 19 VM).

N_{MC}	split (ms)	process (ms)	merge (ms)	total (min)	serial (min)	speed-up
256	—	111 998	2 250	1.9	1.9	1.0
1 024	78	123 487	7 094	2.2	8.0	3.7
16 384	500	451 397	11 875	7.7	127.3	16.5
65 536	2 328	1 576 711	27 031	26.7	509.3	19.0
262 144	7 609	6 078 757	89 450	102.9	2 037.0	19.8
1 048 576	37 766	24 422 238	336 799	413.3	8 148.1	19.7

Analysis of the parallelization gains

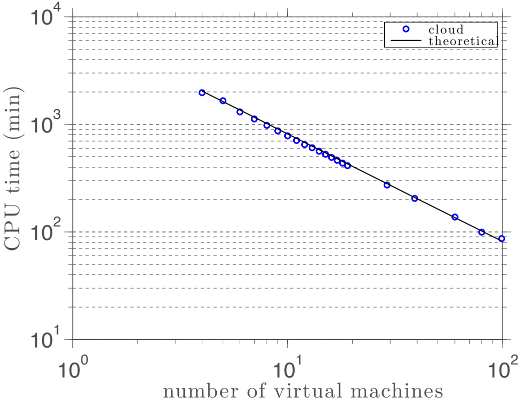


Figure: CPU time of a MC simulation, with 1 048 576 realizations, as function of the numbers of virtual machines.

Costs analysis of the storage space usage

Table: Storage space used by each MC simulations in the cloud, and the financial cost associated.

N_{MC}	space (MB)	cost ($US\$$)
1 024	1.0	1.79
16 384	12.4	5.39
65 536	49.0	5.39
262 144	195.2	7.67
1 048 576	780.8	19.07

Concluding remarks

- A cloud computing setting for MC parallelization is presented
- Estimations of mean and variance show good accuracy
- McCloud requires low storage space for MC simulation
- McCloud obtains large performace gains in MC simulation
- McCloud offers high scalability and low-cost for MC simulation

Acknowledgments

Financial support:

- CNPq
- CAPES
- FAPERJ

Free access to Windows Azure facilities:

- Microsoft Coorporation

References

- A. Cunha Jr , R. Nasser, R. Sampaio , H. Lopes, and K. Breitman **Uncertainty Quantification in a Cloud Computing Setting**. under review in CPC

- R. Nasser, A. Cunha Jr, H. Lopes, K. Breitman, and R. Sampaio **McCloud: Easy and Quick Way to Run Monte Carlo Simulations in the Cloud**. to be submitted to JSS

PUC RIO