

## **COMPARATIVE STUDY BETWEEN TWO VERSIONS OF METROPOLIS-HASTING ALGORITHM FOR GENERATING COMPUTER EXPERIMENT DESIGNS ACCORDING TO A POINT PROCESS**

*Hichem ELMOSSAOUI, Nadia OUKID*

Department of Mathematics, Faculty of Science, University Blida 1,  
Soumaa, B.P 270, Blida, Algeria

### **ABSTRACT**

This work presents the study of the impact, to use the standard approach of the Metropolis-Hasting algorithm and its approach with delayed rejection, on generating computer experiment designs according to the point stochastic model of Marked Strauss. Finally, an application using Matlab has been developed to easily compare between the two approaches.

**Key words :** computer experiment design, Markov chain Monte Carlo (MCMC), Metropolis-Hasting, Marked point process.

### **1. INTRODUCTION**

A simulation consists of setting one or more sets of input variables, performing calculations, and then analysing the results provided by the simulator. Unfortunately, the main difficulties are related to the cost of the calculations from the simulator, and the size of the problem to tackle. Even after a significant reduction in the size of the problem, it is still impossible for some applications to opt for a direct use of the simulator. Hence, the idea is to replace the simulator with one or more approximate functions. These are, generally, simple functions obtained using approximation or interpolation methods, and through computer experiment designs. Among the ways used to generate such computer experiment designs are the simulation techniques with Markov chain (MCMC), and Metropolis-Hastings algorithm (MH).

The idea consists of creating a chain of configurations  $\{X_0, X_1, \dots, X_N\}$  which converge to a desired distribution  $\pi$ . In fact, the Metropolis-Hastings algorithm allows such a construction through a transition nucleus  $P$  which is  $\pi$ -reversible. In this context, it is worth mentioning the works done by Franco J.[1], and Elmoassaoui H. et al [2, 3]. There are many sub-categories of Metropolis-Hastings algorithms [4]. However, the standard Metropolis-Hastings algorithm generally does not work well with high dimensions since it leads to more frequent repeated samplings. Hence, so as to overcome this deficiency we can use another variant of the algorithm called the Metropolis-Hasting algorithm and its approach with delayed rejection (MHRD) which was proposed by Mira in 1999[5]. The key idea behind the Metropolis-Hasting algorithm and its approach with delayed rejection is to reduce the correlation between the states of the Markov chain.

### **2. SIMULATION OF POINT PROCESSES USING MCMC AND GENERALISED METROPOLIS-HASTING ALGORITHM**

Metropolis-algorithm was introduced by Metropolis and al. in 1953 [6], and generalized by Hasting in 1970 [4]. It started to be adopted for the case of spacial processes by Geyer and

Moller way back in 1994 [7]. The basic idea of the algorithm consists of proposing a new state through a slight perturbation of the current state, and then see whether to accept or reject it. Generally, the algorithm uses a transition  $P$  which is  $\pi$ -reversible and  $\pi$ -invariant, and goes through two steps as described below :

1. A change of state from  $x$  to  $y$  according to a distribution  $Q(x, \cdot)$  is proposed,
2. State  $y$  is accepted with the probability  $a(x, y)$ , else state  $x$  shall be kept (where  $a : \Omega \times \Omega \rightarrow [0, 1]$   $a : \Omega \times \Omega \rightarrow [0, 1]$ )

Let  $q(x, y)$  denote the density of  $Q(x, \cdot)$ , then the transition  $P$  of MH is given as follows [8] :

$$P_{MH}(x, y) = a(x, y)q(x, y) + \left[ 1 - \int_{\Omega} a(x, z)q(x, z) dz \right] \delta_x(y),$$

Where  $\delta_x(\cdot)$  is the mass at point  $x$ . To make calculations simple, we use Dirac measurement at  $x$  ( $\delta_x(y) = 1$  if  $x = y$ , and 0 otherwise).

The choice of  $(Q, a)$  guarantees the  $\pi$ -reversibility of  $P_{MH}$  provided the following equilibrium equation is satisfied :

$$\forall x, y \in \Omega : \pi(x) \times q(x, y) \times a(x, y) = \pi(y) \times q(y, x) \times a(y, x)$$

The choice of the acceptance probability  $a$  is more limited, it is mainly driven by the aim to simulate (asymptotically) a given distribution  $\pi$ . It is the case with the usual choice where :

$$a(x, y) = \frac{\pi(y) \times q(y, x)}{\pi(x) \times q(x, y)}$$

### 3. SIMULATION OF POINT PROCESSES USING MCMC METHOD AND METROPOLIS-HASTING ALGORITHM WITH DELAYED REJECTION

The Metropolis-Hasting algorithm proposes one single candidate at each iteration. The MH algorithm with delayed rejection [5] is an example of an algorithm which considers many candidates at each iteration. Thus, a maximal number  $k$  of candidates are chosen ; at each iteration, a candidate is proposed either until it is accepted, or until the maximal number is reached. For example : for  $k = 2$ , the acceptance probability of the first candidate  $y_1$  is the same as that of the generalized MH algorithm ; then that of the second candidate  $y_2$  is given as follows :

$$a_2(x, y_1, y_2) = \min \left( \frac{\pi(y_2) q_1(y_2, y_1) q_2(y_2, y_1, x) [1 - a_1(y_2, y_1)]}{\pi(x) q_1(x, y_1) q_2(x, y_1, y_2) [1 - a_1(x, y_1)]}, 1 \right)$$

where  $q_1$  and  $q_2$  are the instrumental densities of the first and second candidates respectively.

### 4. . COMPARISON RESULTS

So as to judge the quality of a computer experiment design, it is important to opt for usual criteria which allow good filling of the space, and a good uniform distribution. The aim of this section is to calculate the values of these criteria on marked Strauss designs [2], which are generated using the two versions of Metroplois-Hasting algorithm.

So as to make the results meaningful, the following three types of criteria are then used

- Distance criterion [9],
- Recovery criterion (Coefficient R) [10],
- Discrepancy criterion [11].

The figures below represent the box-plots obtained as a result of this work using the two versions of the algorithm; studied and applied for the case of many dimensions. We have opted for the standard normal distribution as a random variable generating engine for the two versions as well as for  $q_1$  and  $q_2$ .

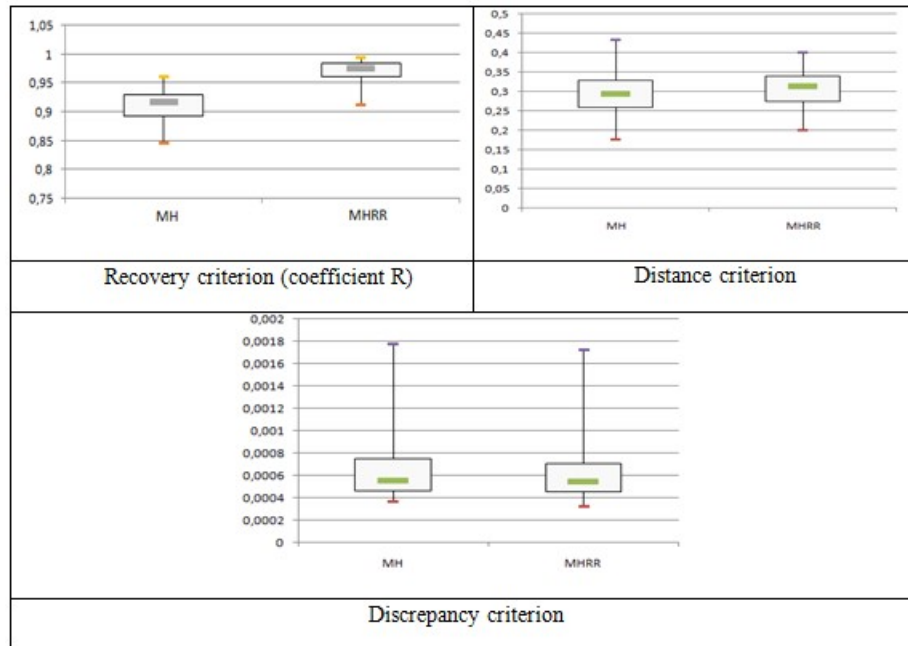


FIGURE 1 – Boxplots of distance, discrepancy, and recovery quality criteria calculated on the basis of 40 designs of 50 points for dimension 5.

A quick analysis of the results shown in the figure above reveals that Metropolis-Hasting algorithm with delayed rejection offers, in most cases, the best results with regard to the three criteria involved in this comparison.

## 5. CONCLUSION

The use of MCMC methods and Metropolis-Hasting algorithm in the context of computer experiment designs allows to build new specified designs using some distribution. This approach provides greater flexibility since we can easily manipulate this distribution through its representation so as to impose some properties such as the filling of the study domain.

Following the comparison done in the course of this work, we have found out that change proposed by Metropolis-Hasting algorithm with delayed rejection tackles the problem of the same state being repeated for the chain when the value given by the general algorithm is rejected. Intuitively, the fact of sticking to the same state most of the times suggests that the algorithm has failed to explore all of the space, and that the correlation between the values increases. Moreover, information on the value rejected in the regular Metropolis-Hasting algorithm cannot be used subsequently for the sake of preserving Markov chain properties. On the other hand,

since acceptance probabilities at each state should be (separately) adjusted in Metropolis-Hasting algorithm with delayed rejection in order to preserve the reversibility property of the chain, the algorithm, then, allows to use the information acquired through rejections within the same iteration ; hence offering the possibility of a local adjustment of the instrumental distribution.

## 6. REFERENCES

- [1] Franco, J. “Planification d’Expériences Numériques en Phase Exploratoire pour des Codes de Calculs Simulant des Phénomènes Complexes”, Doctoral thesis, l’Ecole Nationale Supérieure des Mines de Saint-Etienne, France (2008).
- [2] Elmoosaoui, H., Oukid, N. and Hananne, F “Construction of computer experiment designs using marked point processes”, *Afrika Matematika*, Doi.org /10.1007/ s13370-020-00770-9, Springer, (2020).
- [3] Elmoosaoui, H., “contribution a la méthodologie de la recherche expérimentale ” , thèse de doctorat, Université Blida1 (2020).
- [4] Hastings, W.K., “Monte Carlo sampling methods using markov chains and their applications”, *Biometrika* 57, (1970), 97–109.
- [5] Mira, A., “On metropolis hasting algorithm with delayed rejection, *Metron*”, Vol LIX, n.3-4, (200) 231-241
- [6] Geyer, C.J. and Moller, J., “Simulation and likelihood inference for spatial point processes”, *Scandinavian Journal of Statistics, Series B*, 21 :359–373, (1994).
- [7] Geyer, C.J. and Moller, J., “Simulation and likelihood inference for spatial point processes”, *Scandinavian Journal of Statistics, Series B*, 21 :359–373, (1994).
- [8] Meyen, S.P. and Tweedie, R.L., “Markov chains and stochastic stability”, springer, New York, (1993).
- [9] Johnson, M, E., Moore, L.M., Ylvisaker, D. “ Minimax and Maximin Distance Designs. *Journal of Statistical Planning and Inference*”, 26 : 131-148(1990).
- [10] Gunzburger, M., Burkardt, J. “ Uniformity measures for point samples in hypercubes”, [www.csit.fsu.edu/burkardt/pdf/ptmeas.pdf](http://www.csit.fsu.edu/burkardt/pdf/ptmeas.pdf).
- [11] Warnock T.T. “Computational Investigations of Low-Discrepancy Point Sets II”, In : Niederreiter H., and Shiue P.J.S. (eds) *Monte Carlo and Quasi-Monte Carlo Methods in Scientific Computing*. Lecture Notes in Statistics, 106. Springer, New York (1995).