



Statistical Theory and Related Fields

ISSN: (Print) (Online) Journal homepage: https://www.tandfonline.com/loi/tstf20

# Bayesian autoregressive adaptive refined descriptive sampling algorithm in the Monte Carlo simulation

Djoweyda Ghouil & Megdouda Ourbih-Tari

**To cite this article:** Djoweyda Ghouil & Megdouda Ourbih-Tari (2023) Bayesian autoregressive adaptive refined descriptive sampling algorithm in the Monte Carlo simulation, Statistical Theory and Related Fields, 7:3, 177-187, DOI: <u>10.1080/24754269.2023.2180225</u>

To link to this article: https://doi.org/10.1080/24754269.2023.2180225

© 2023 The Author(s). Published by Informa UK Limited, trading as Taylor & Francis Group.



Published online: 15 May 2023.

Submit your article to this journal  $\square$ 

Article views: 186



View related articles 🗹

🛛 View Crossmark data 🗹



Taylor & Francis

OPEN ACCESS Check for updates

# Bayesian autoregressive adaptive refined descriptive sampling algorithm in the Monte Carlo simulation

# Djoweyda Ghouil<sup>a,b</sup> and Megdouda Ourbih-Tari<sup>c,d</sup>

<sup>a</sup>Département de Mathématiques, Faculté des Sciences Exactes, Université de Tizi Ouzou, Tizi Ouzou, Algeria; <sup>b</sup>Laboratoire de Mathématiques appliquées, Faculté des Sciences Exactes et Informatique, Université de Jijel, Jijel, Algeria; <sup>c</sup>Institute of Science, University Center of Tipaza, Tipaza, Algeria; <sup>d</sup>Laboratoire de Mathématiques appliquées, Faculté des Sciences Exactes, Université de Bejaia, Bejaia, Algeria

#### ABSTRACT

This paper deals with the Monte Carlo Simulation in a Bayesian framework. It shows the importance of the use of Monte Carlo experiments through refined descriptive sampling within the autoregressive model  $X_t = \rho X_{t-1} + Y_t$ , where  $0 < \rho < 1$  and the errors  $Y_t$  are independent random variables following an exponential distribution of parameter  $\theta$ . To achieve this, a Bayesian Autoregressive Adaptive Refined Descriptive Sampling (B2ARDS) algorithm is proposed to estimate the parameters  $\rho$  and  $\theta$  of such a model by a Bayesian method. We have used the same prior as the one already used by some authors, and computed their properties when the Normality error assumption is released to an exponential distribution. The results show that B2ARDS algorithm provides accurate and efficient point estimates.

#### **ARTICLE HISTORY**

Received 13 May 2021 Revised 9 December 2022 Accepted 2 February 2023

#### **KEYWORDS**

Monte Carlo simulation; refined descriptive sampling methods; variance reduction; autoregressive process; Bayesian estimation

# 1. Introduction

Time series are often used to model data in different areas of life. We can think of economic forecasts, the evolution of a biological system, time-stamped data from social media, E-commerce and sensor systems. The AutoRegressive (AR) model is the most widely used class of time series models, and particularly the first order AutoRegressive process denoted as AR(1).

On the other hand, the use of Bayesian inference has developed rapidly over the last decade in several fields including medicine (Ashby, 2006), genetics (Beaumont & Rannala, 2004), bioinformatics (Corander, 2006), econometrics (Geweke, 1999), etc.

Several works related to time series models have been treated by the Bayesian approach. Box et al. (1976), Monahan (1984) and Marriot and Smith (1992) discussed the Bayesian approach to analyse the AR models. Diaz and Farah (1981) devoted their work to a Bayesian technique for computing posterior analysis of an AR process in any arbitrary order. Amaral Turkmann (1990) considered a Bayesian study of an AR(1) model with exponential white noise based on a noninformative prior. Ghosh and Heo (2003) introduced a comparative study to some selected noninformative priors for the AR(1) model. Pereira and Amaral-Turkman (2004) developed a Bayesian analysis of a threshold AR model with exponential white noise.

Several Bayesian methods for estimating autoregressive parameters have been proposed by different authors; for instance, Albert and Chib (1993) and Barnett et al. (1996) proposed a Bayesian method via Gibbs sampling to estimate the parameters in an AR model. Ibazizen and Fellag (2003) proposed a Bayesian estimation of an AR(1) process with exponential white noise by using a noninformative prior, whereas Suparman and Rusiman (2018) proposed a hierarchical Bayesian estimation for stationary AR models using a reversible jump MCMC algorithm. However, Kumar and Agiwal (2019) proposed a Bayesian estimation for a full shifted Panel AR(1) time series model.

Bayesian inference is based on the posterior distribution, which can be interpreted as the result of the combination of two information sources: the information provided by the parameter which is contained in the prior distribution, and the information provided by the data which formalizes the likelihood function through the use of a given statistical model.

Posterior inference can be undertaken by applying classical Monte Carlo (MC) methods, which provide iterative algorithms that can generate approximate samples from the posterior distribution. These methods fall into two categories: the Markov Chain Monte Carlo (MCMC) methods such as Metropolis-Hastings and Gibbs algorithms (Smith & Robert, 1993), and the Monte Carlo (MC) method also known as Simple Random Sampling (SRS), for

CONTACT Megdouda Ourbih-Tari 🖾 ourbih.megdouda@cu-tipaza.dz 💿 Institute of Science, University Center of Tipaza, 42020 Tipaza, Algeria; Laboratoire de Mathématiques appliquées, Faculté des Sciences Exactes, Université de Bejaia, Bejaia 06000, Algeria

© 2023 The Author(s). Published by Informa UK Limited, trading as Taylor & Francis Group.

This is an Open Access article distributed under the terms of the Creative Commons Attribution License (http://creativecommons.org/licenses/by/4.0/), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited. The terms on which this article has been published allow the posting of the Accepted Manuscript in a repository by the author(s) or with their consent.

		1	•	112				
i	$r_i^1$	<i>x</i> <sup>1</sup> <sub><i>i</i></sub>	i	$r_i^2$	<i>x</i> <sup>2</sup> <sub><i>i</i></sub>	i	$r_i^3$	<i>x</i> <sup>3</sup> <sub><i>i</i></sub>
1	0.071	-0.970	1	0.045	-1.129	1	0.038	-1.181
2	0.214	-0.432	2	0.136	-0.689	2	0.115	-0.770
3	0.357	-0.029	3	0.227	-0.393	3	0.192	-0.501
4	0.500	0.367	4	0.318	-0.136	4	0.269	-0.272
5	0.643	0.817	5	0.409	0.112	5	0.346	-0.059
6	0.786	1.422	6	0.500	0.367	6	0.423	0.151
7	0.929	2.602	7	0.591	0.642	7	0.500	0.367
	$\hat{\theta}_1$	0.539	8	0.682	0.960	8	0.577	0.601
			9	0.773	1.355	9	0.654	0.856
			10	0.864	1.920	10	0.731	1.159
			11	0.955	3.068	11	0.808	1.544
				$\hat{\theta}_2$	0.553	12	0.885	2.099
						13	0.962	3.239
							$\hat{ heta}_3$	0.556

**Table 1.** Subsets of prime numbers size for a Gumbel distribution with mean E(X) = 0.577,  $p_1 = 7$ ,  $p_2 = 11$  and  $p_3 = 13$ .

**Table 2.** The observed mean of a Gumbel distribution using three regular subsets of prime number size.

j	1	2	3	Overall mean (T <sub>RDS</sub> )
θ <sub>j</sub>	0.539	0.553	0.556	0.551

instance, Quasi Monte Carlo (Keller et al., 2008), Latin hypercube sampling (LHS) (Loh, 1996; Mckay et al., 1979; Stein, 1987), Descriptive Sampling (DS) (Saliby, 1990) as well as Refined Descriptive Sampling (RDS) (Tari & Dahmani, 2006). In the non-Bayesian estimation, the SRS is a widely used method in MC simulation. But unfortunately, the sampling method generates estimation errors, so it is not accurate enough. However, the RDS method (Aloui et al., 2015) has shown its effectiveness for several models and in various fields. For instance, in the field of medicine, Ourbih-Tari and Azzal (2017) proposed an RDS algorithm for a nonparametric estimation of the survival function using Kaplan Meier and Fleming Harrington estimators in a case study. In the field of engineering, Baghdali-Ourbih et al. (2017) have estimated the performance measures of the aggregate quarry of Bejaia city in Algeria, using a discrete-event simulation model. In the field of M/G/1 queuing models, M/M/1 retrial queues and M/G/1 retrial queues, simulation results using RDS outperform those obtained using the MC method (Idjis et al., 2017; Ourbih-Tari et al., 2017). Also, a comparison between RDS and LHS carried out on M/G/1 queues showed the efficiency of RDS over LHS (Boubalou et al., 2019). Since it is important to choose a sampling scheme that allows for a better parameter estimate while keeping the number of repetitions to a minimum, the RDS method is chosen for further work.

This paper will therefore explore the RDS method in a Bayesian sense by considering the AR(1) model, which has already been studied by Amaral Turkmann (1990) and, later on, by Ibazizen and Fellag (2003). For this purpose, we assume the noninformative prior given in Ibazizen and Fellag (2003) for the AR(1) process with exponential white noise. Then, we develop an algorithm for the estimation of AR(1) model parameters in a Bayesian context by using the most accurate sampling method as the RDS method. The proposed algorithm is called Bayesian Autoregressive Adaptive Refined Descriptive Sampling (B2ARDS).

Some details of the RDS method are given in Section 2. Then, in Section 3, the methodology from Bayesian perspective is explained. Section 4 is concerned with the Bayesian estimates of the AR(1) model parameters and their variances using the RDS method through the B2ARDS algorithm. Section 5 presents simulation experiments performed by using the R Package and corresponding results are summarized in Tables 1 and 2, and Figures 1 and 2. Finally, Section 6 concludes this paper.

# 2. The use of RDS

DS is known to have two problems. From a theoretical perspective, DS can be biased. More practically, its strict operation requires a prior knowledge of the required sample size. Based on this, RDS has been proposed to make DS safe, efficient and convenient. RDS makes DS safe by reducing substantially the risk of sampling bias, efficient by producing estimates with lower variances and convenient by removing the need to determine in advance the sample size. Not only this, but RDS beats DS in several comparisons and such is the case, on a flow shop system (Tari & Dahmani, 2005a) and a production system (Tari & Dahmani, 2005b). So, RDS is the preferred method when it comes to performing MC simulation.



**Figure 1.** Sequences of  $\hat{\rho}_i$  estimates for different true values of  $\rho$ . (a)  $\rho = 0.1$ . (b)  $\rho = 0.3$ . (c)  $\rho = 0.6$ .

#### 2.1. Description of the method

RDS is based on a block of sub-samples of randomly generated prime number sizes, so it distributes RDS sample values from sub-samples at the request of the simulation. We stop the process of the generation of sample values when the simulation terminates, say when *m* prime numbers  $p_j$ , j = 1, 2, ..., m, have been used, which derives *m* sub-runs.

The choice of primes in the RDS method as sub-sample sizes is motivated by the definition of primes since we must avoid integers having multiples, whether they are even or odd numbers. Indeed, if the underlying frequency is different from the sampling frequency (the product of primes) or a multiple of it, then the bias of the estimates will be insignificant as demonstrated in Tari and Dahmani (2006). This should ensure that the method reduces the sampling bias.

**Remark:** In the case of *M* replicated simulation runs, the RDS method will distribute *M* blocks defined by  $m_1, m_2, \ldots, m_M$  primes which are not the same for all replicated simulation runs.

# 2.2. RDS sample values

Formally, in an RDS run defined by a sample of size  $n = \sum_{j=1}^{m} p_j$ , the sample values drawn from the input real-valued random variable *X* having *F* as the cumulative distribution are given by the following formulas:

$$x_i^j = F^{-1}\left(R_i^j\right)$$
 for  $i = 1, 2, \dots, p_j$  and  $j = 1, 2, \dots, m$ , (1)

where for any j = 1, 2, ..., m, the values of each sub-sample  $(R_i^j, i = 1, 2, ..., p_j)$  are randomly selected without replacement from the regular numbers of the following sub-set

$$\left(r_{i}^{j} = \frac{(i-0.5)}{p_{j}}, \quad i = 1, 2, \dots, p_{j}\right)$$
 (2)



**Figure 2.** Sequences of  $\hat{\theta}_i$  estimates for  $\theta = 2.5$  and different values of  $\rho$ . (a)  $\rho = 0.1$ . (b)  $\rho = 0.3$ . (c)  $\rho = 0.6$ .

and such  $r_i^j$  is the midpoint of the  $p_j$  subintervals in which the interval [0, 1] is subdivided. The primes  $p_j$ , j = 1, 2, ..., m, are randomly generated between 7 and a given value fixed by the user.

# 2.3. The estimator of RDS

The *j*-th sub-run generating  $p_j$  values of the input random variable X leads to the following estimate of the parameter  $\theta$ 

$$\hat{\theta}_j = \frac{1}{p_j} \sum_{i=1}^{p_j} h(x_i^j).$$
(3)

Therefore, in a given run, the use of refined descriptive sample of size  $n = \sum_{j=1}^{m} p_j$  leads to the following sampling estimate of  $\theta$  defined by the average of those estimates observed on different sub-runs:

$$T_{\rm RDS} = \frac{1}{n} \sum_{j=1}^{m} \sum_{i=1}^{p_j} h(x_i^j), \tag{4}$$

where h is the simulation function.

#### 2.4. Example

We illustrate the method by taking the Gumbel distribution and the prime numbers  $p_1 = 7$ ,  $p_2 = 11$  and  $p_3 = 13$ . The values of the variable are simulated by the formula  $x_i = -\ln(-\ln(r_i))$ ,  $i = 1, 2, ..., p_j$  and j = 1, 2, ..., m. When m = 3, the obtained sub-sets and sub-samples are given in Table 1 using the formulas (2) and (1), respectively, together with the observed mean  $(\hat{\theta}_j)$  in each sub-run using the formula (3), while the overall mean computed at the end of the run is given in Table 2 using the formula (4).

# 3. Monte Carlo methods from a Bayesian perspective

We consider a random vector *Y* having a probability density function  $f(y | \theta)$ , where  $\theta$  is a vector of unknown parameters. The Bayesian statistic considers  $\theta$  as a random vector of density  $\pi(\theta)$  called prior law of  $\theta$ .

In a Bayesian sense, any inference is based on the computation of the posterior distribution which is the conditional distribution of the random vector  $\theta$  knowing y and is denoted by its probability density  $\pi(\theta \mid y)$ . Bayesian estimator of the parameter  $\theta$  is constructed from the posterior distribution  $\pi(\theta \mid y)$  by minimizing an appropriate loss function. Most Bayesian estimators have the form of  $E_{\pi}[h(\theta)]$ , where  $E_{\pi}$  denotes the mathematical expectation of the posterior distribution  $\pi(\theta \mid y)$ .

A major difficulty in an estimation by the Bayesian approach, is the explicit computation of the posterior distribution as soon as the number of parameters exceeds two. Also computing  $E_{\pi}[h(\theta)]$  can be very difficult even if the law is known. Thus, several approximation methods such as the MC and MCMC methods are proposed in the literature. The latter allows for the estimation of moments of any order of random variables, too complex to be studied analytically, using iterative generation algorithms.

The Monte Carlo method is a numerical method of solving problems using random numbers. It is one of the most popular mathematical tools with a very wide scope, like differential equation integration, fluid mechanics equations, financial mathematics, particles transport and matrix inversion.

In a Bayesian sense, the Monte Carlo method consists in estimating

$$I_{h} = E_{\pi}[h(\theta)] = \int_{\theta} h(\theta)\pi(\theta \mid y) d\theta$$

by

$$\hat{I}_h = \frac{1}{n} \sum_{i=1}^n h(\theta_i),$$

where  $\theta_1, \theta_2, \dots, \theta_n$  are independent and identically distributed i.i.d. random variables according to  $\pi(\theta \mid y)$ . The strong law of large numbers ensures that  $\hat{I}_h$  converges almost surely to  $I_h$  when *n* tends to infinity.

#### 4. The use of RDS for the simulation of the proposed model

#### 4.1. Presentation of the model

Consider the first order autoregressive process defined by

$$X_t = \rho X_{t-1} + Y_t, \quad t = \dots, -1, 0, 1, \dots,$$
 (5)

where  $0 < \rho < 1$  and the  $Y_t$ 's are i.i.d. random variables with the exponential distribution  $\exp(\theta)$  of parameter  $\theta$  having a density

$$f(y) = \theta \exp(-\theta y) I_{(0,\infty)}(y), \quad \theta > 0.$$

 $X_1$  is assumed to be distributed according to  $\exp((1-\rho)\theta)$  such that the process is mean stationary.

The likelihood function based on the observations  $x = \{x_1, x_2, ..., x_n\}$  is

$$f(x \mid \theta, \rho) = (1 - \rho)\theta^{n} e^{-\theta(n\bar{x} - \rho S)} I_{A}(x),$$

where

$$A = \{x : x_1 > 0, x_t - \rho x_{t-1} \ge 0, t = 2, \dots, n\}, \quad n\overline{x} = \sum_{i=1}^n x_i$$

and

$$S = n\overline{x} - (x_n - x_1). \tag{6}$$

The Maximum Likelihood Estimators (MLE) of  $\rho$  and  $\theta$  are introduced by Andel (1988), respectively as follows:

$$\rho_0 = \min\left(1, \frac{x_2}{x_1}, \dots, \frac{x_n}{x_1}\right),\tag{7}$$

$$\theta_0 = \frac{n}{n\bar{x} - \rho_0 S}.$$
(8)

The model given by (5) was processed using a Bayesian approach by Amaral Turkmann (1990). The author's study is based on a noninformative prior defined as follows:

$$\pi(\theta,\rho) \propto \frac{1}{\theta(1-\rho)} I_{[0,\infty]\times[0,1)}(\theta,\rho).$$
(9)

The author derived the Bayes estimators of the independent parameters  $\rho$  and  $\theta$  as follows:

$$\rho^{B} = \frac{\rho_{0}}{n-2} \left( \frac{n-1}{1-r^{n-1}} - \frac{1}{1-r} \right),$$

and

$$\theta^B = \frac{n-1}{n\overline{x}r} \frac{1-r^n}{1-r^{n-1}},$$

where  $r = 1 - \rho_0(\frac{S}{n\overline{x}})$  such that 0 < r < 1.

Ibazizen and Fellag (2003) proposed a Bayesian analysis of the model (5) by using a more general noninformative prior defined by

$$\pi(\theta,\rho,\beta) \propto \frac{1}{\theta} \frac{\rho^{\beta-1}}{1-\rho} I_{[0,\infty]\times[0,1)}(\theta,\rho), \quad \beta > 0$$
(10)

which includes the formula (9) as a special case ( $\beta = 1$ ). For  $\theta$ , the prior (7) is proportional to the scale parameter  $1/\theta$  and for  $\rho$ , it is a special case of the beta distribution.

The authors have derived Bayesian estimators given below, but their formulas are complicated for computation. So, simulation experiments were carried out and the obtained results were compared to the MLE.

$$\hat{\rho}_B(\beta) = \frac{{}_2F_1(\beta + 1, n, \beta + 2; 1 - r)\rho_0\beta}{{}_2F_1(\beta, n, \beta + 1; 1 - r)(\beta + 1)}$$
$$\hat{\theta}_B(\beta) = \frac{{}_2F_1(\beta, n + 1, \beta + 1; 1 - r)}{\bar{x} {}_2F_1(\beta, n, \beta + 1; 1 - r)},$$

where

$${}_{2}F_{1}(a,b,c;x) = \sum_{m=0}^{\infty} \frac{(a,m)(b,m)x^{m}}{(c,m)m!}$$
$$= \int_{0}^{1} \frac{u^{a-1}(1-u)^{c-a-1}(1-xu)^{-b}}{\mathcal{B}(a,c-a)} du \qquad \text{for } |x| < 1$$

is the Gauss hypergeometric function of parameters (a, b, c).

We note that to compute the latter, the Pochhammer coefficients (t, m) for t = a, b and c are defined as follows

$$(t,m) = \Gamma(t+m)/\Gamma(t)$$
 for  $m \ge 1$  and  $(t,0) = 1$ ,

where  $\Gamma$  and  $\mathcal{B}$  represent symbols of the usual functions Gamma and Beta, respectively.

This paper focuses on the Bayesian estimation of  $\rho$  and  $\theta$  using the most accurate RDS method to generate numbers for the simulation, where the prior distribution of  $\rho$  and  $\theta$  is the one given in (10) because it is the most general noninformative prior.

#### **4.2.** Bayesian estimation of $\rho$ and $\theta$ using the RDS method

Suppose that our data consists of the first *n* observations  $x = (x_1, x_2, ..., x_n)$ .

By using the prior in (10) and the Bayes theorem, we get the following posterior:

$$\pi(\theta, \rho \mid x) = \frac{\rho^{\beta - 1} \theta^{n - 1} \mathrm{e}^{-\theta(nx - \rho S)} I_{(0, \infty) \times [0, \rho_0]}(\theta, \rho)}{\Gamma(n) \int_0^{\rho_0} \rho^{\beta - 1} (n\overline{x} - \rho S)^{-n} \mathrm{d}\rho},\tag{11}$$

which is proportional to

$$\rho^{\beta-1}\theta^{n-1} \mathrm{e}^{-\theta(n\bar{x}-\rho S)} I_{(0,\infty)\times[0,\rho_0]}(\theta,\rho).$$
(12)

Knowing that the parameters  $\rho$  and  $\theta$  are independent, the conditional posterior distributions of  $\rho$  and  $\theta$  are respectively

$$F(\rho) = \pi(\rho \mid x, \theta) \propto \rho^{\beta - 1} e^{\theta \rho S}$$
(13)

and

$$H(\theta) = \pi(\theta \mid x, \rho) \propto \theta^{n-1} e^{-\theta(n\bar{x} - \rho S)}.$$
(14)

The use of RDS method requires the computation of the inverse cumulative distribution functions  $F^{-1}$  and  $H^{-1}$ , so we put

$$R_1 = F(\rho) = \rho^{\beta - 1} e^{\theta \rho S},$$
  

$$R_2 = H(\theta) = \theta^{n - 1} e^{-\theta (n\bar{x} - \rho S)}$$

The inversion method leads to the following  $\rho$  and  $\theta$  values of the conditional posterior distributions which will be used for the generation of observations:

$$\rho = F^{-1}(R_1) = \frac{1}{\theta S} \ln\left(\frac{R_1}{\rho^{\beta - 1}}\right) \tag{15}$$

and

$$\theta = H^{-1}(R_2) = \frac{1}{n\overline{x} - \rho S} \ln\left(\frac{\theta^{n-1}}{R_2}\right),\tag{16}$$

where the variables  $R_1$  and  $R_2$  following  $\mathcal{U}[0, 1]$  are generated using the RDS algorithm given in subsection 4.3 (see Appendix 1 for the generation of the  $\rho$  values and Appendix 2 for the generation of the  $\theta$  values).

#### 4.3. Adaptive RDS algorithm to AR(1) model from the Bayesian perspective by using the R language

In this section, we present a new Bayesian Adaptive AR(1) RDS (B2ARDS) algorithm to estimate the parameters of the model given in (2).

## Data structure

For each input random variable, we define a record with the following structure.

*N*: an integer defining the number of replications.

*n*: an integer defining the sample size.

*p*: a prime number defining the size of the sub-sample.

*P*: an array of generated primes.

*r*: an array  $[1, \ldots, p]$  of real numbers containing the sub-set of regular numbers.

*R*: an array of mixed values of *r*.

 $x_i$ , i = 1, ..., n: simulated observations from the AR(1) model.

 $\rho$ : an array  $[1, \ldots, p]$  containing the generated RDS sub-sample values, issued from the  $\rho$  distribution.

 $\theta$ : an array  $[1, \ldots, p]$  containing the generated RDS sub-sample values, issued from the  $\theta$  distribution.

# Algorithm

- (a) Initialization for the experiment of *N* replicated runs.
- (b) Initialization of the run defined by the sample size *n*.
- (c) Simulate samples of *n* observations  $x_1, x_2, ..., x_n$  from the model (5) by using the command  $: > x = arima.sim(list(ar = <math>\rho), n, innov = rexp(n, rate = \theta))$ .

184 😉 D. GHOUIL AND M. OURBIH-TARI

- (d) Compute the values  $\overline{x}$  and S, using formula (6) and  $n\overline{x} \rho S$ .
- (e) Before each simulation run, generate a vector of prime numbers smaller than the sample size *n* and store them in a vector *P*.
- (f) Randomize the vector P.
- (g) Compute the sum of primes successively and stop when the sum is  $\ge n$  and collect the number of primes *m*.
- (h) Initialization of the sub-run defined by the prime *p*.
- (i) For each prime number  $p_j$ , j = 1, ..., m, we do the following sub-run.
  - (*f*<sub>1</sub>) Compute the regular numbers  $r_i$ ,  $i = 1, ..., p_j$  using the formula (2) and store them in an array r.
  - (*f*<sub>2</sub>) Randomize the  $r_i$  by the command: > R = sample(r) and store them in an array R.
  - (f<sub>3</sub>) Generate the observations  $\rho_i = F^{-1}(R_i)$  and  $\theta_i = H^{-1}(R_i)$ ,  $i = 1, ..., p_j$  and store them respectively in arrays  $\rho$  and  $\theta$ .
  - (*f*<sub>4</sub>) Compute the simulation estimates  $\hat{\rho}_i$  and  $\hat{\theta}_j$  in the *j*-th sub-run such that

$$\hat{\rho}_j = \frac{1}{p_j} \sum_{i=1}^{p_j} \rho_i$$
,  $\hat{\theta}_j = \frac{1}{p_j} \sum_{i=1}^{p_j} \theta_i$ .

(j) Compute the final RDS estimates of  $\rho$  and  $\theta$  using the output sample of size  $\sum_{i=1}^{m} p_i$  respectively as

$$\hat{\rho}_{\text{RDS}} = \frac{\sum_{j=1}^{m} p_j \hat{\rho}_j}{\sum_{j=1}^{m} p_j}, \quad \hat{\theta}_{\text{RDS}} = \frac{\sum_{j=1}^{m} p_j \hat{\theta}_j}{\sum_{j=1}^{m} p_j}$$

# 5. Results and discussion

#### 5.1. Simulation study

In the following, we present a simulation study using the R language. We generate samples of *n* observations  $x_1, x_2, \ldots, x_n$  from the AR(1) model given in (5) which provides the values of  $\overline{x}$ , S and  $n\overline{x} - \rho S$ . Three simulation experiments were carried out with different sample sizes such as n = 20, 30, 40 for each selected  $\rho$  value such as 0.1, 0.3, 0.6, respectively. The proposed B2ARDS algorithm is used to compute the RDS estimates  $\hat{\rho}$  and  $\hat{\theta}$  together with their variances. Note that, in the estimation of  $\rho$ , different values of  $\beta$  are taken from one experiment to another in order to study the effect of varying  $\beta$  on the Bayesian estimates  $\hat{\rho}$  and  $\hat{\theta}$ . A summary of results is given in Tables 1 and 2, together with  $\rho_0$  and  $\theta_0$ , computed using formulas (7) and (8), respectively (see Appendix 3 for details).

At the end of the simulation, we collect the sequences of  $\hat{\rho}_j$  and  $\theta_j$  estimates resulting from step ( $f_4$ ) of the proposed algorithm. For N = 10,000 replications, sequences of  $\hat{\rho}_j$  estimates for initial values of  $\rho = 0.1, 0.3, 0.6$  are represented in Figure 1 while Figure 2 represents sequences of  $\hat{\theta}_j$  estimates for the same  $\rho$  values as in Figure 1.

#### 5.2. Interpretation of tables

According to the simulation results given in Tables 3 and 4, we have very small variances (given in parentheses). So, we can say that the sequences of RDS values  $\rho_i$  and  $\theta_i$ , generated by the proposed B2ARDS algorithm are homogeneous. Also, in each case  $\hat{\rho}_{RDS}$  is too close to the true value of  $\rho$  while, for some values of  $\beta$ ,  $\hat{\rho}_{RDS}$  is closer to the true value than  $\rho_0$ . It can be seen in Figure 1 that when  $\rho$  is close to 0, the best values of  $\hat{\rho}_{RDS}$  are obtained when  $\beta \in [2.5, 16]$ ; while, when  $\rho$  is close to 1, the best values of  $\hat{\rho}_{RDS}$  are obtained for high values of  $\beta$  ( $\beta \in [50, 111]$ ). So, there are some values for  $\rho$  and  $\beta$  for which the resulting Bayesian estimate is ideal in accuracy.

For  $\hat{\theta}_{RDS}^B$ , the results in Table 4 and Figure 2 also prove that, for each case, the Bayesian estimate of  $\theta$  is very close to the true value of  $\theta$  and for some cases (n = 20 and  $\rho = 0.6$ ), it is a better estimate than  $\theta_0$ . We notice here that the best values of  $\hat{\theta}_{RDS}^B$  are obtained when  $\rho$  is close to 1.

#### 5.3. Interpretation of figures

In each figure, the red line represents the true value of the parameter, while the green line represents the value of the resulting  $\hat{\rho}_{RDS}$  and  $\hat{\theta}_{RDS}$  estimators for the *j*-th sub-run. We notice that the points which represent the estimates  $\hat{\rho}_j$  and  $\hat{\theta}_j$  are concentrated in a given band for all figures and nearly all points are concentrated around their mean, which represents in a Bayesian sense the resulting  $\hat{\rho}_{RDS}$  and  $\hat{\theta}_{RDS}$  estimators of the parameters  $\rho$  and  $\theta$ , respectively.

**Table 3.** The simulated estimator  $\hat{\rho}_{\text{RDS}}$  and its variance for  $\theta = 2.5$  and for various values of  $\rho$ , *n* and  $\beta$ . Variances in parentheses.

$\rho = 0.1$					
		$\hat{ ho}_{RDS}$			
n	$ ho_0$	$\beta = 2$	$\beta = 2.5$	$\beta = 3$	$\beta = 3.5$
20 30	0.1067	0.0798 (0.0032) 0.0445 (0.00008)	0.1188 (0.0009) 0.0812 (0.0002)	0.1886 (0.0021) 0.1144 (0.0005)	0.2284 (0.0032) 0.1499 (0.0009)
40	0.1062	0.0306 (0.00002)	0.0581 (0.00009)	0.0846 (0.0002)	0.1107 (0.0003)
$\rho = 0.3$					
n 20	ρ <sub>0</sub> 0.3035	eta = 7.5 0.2815 (0.0078)	eta = 8 0.3047 (0.0084)	eta = 8.5 0.3009 (0.0056)	eta = 9 0.3232 (0.0063)
n 30	ρ <sub>0</sub> 0.3030	eta = 10.5 0.2633 (0.0030)	eta = 11 0.2736 (0.0032)	eta = 11.5 0.2947 (0.0034)	eta = 12 0.3034 (0.0036)
n 40	ρ <sub>0</sub> 0.3080	eta = 15 0.2926 (0.0026)	eta = 15.5 0.3006 (0.0026)	eta = 16.0 0.3125 (0.0028)	$\beta = 16.5$ 0.3236 (0.0031)
$\rho = 0.6$					
		$\hat{ ho}_{RDS}$			
n 20	ρ <sub>0</sub> 0.6435	$\beta = 50$ 0.5980(0.0559)	eta = 50.5 0.6017 (0.0516)	eta = 55 0.6191 (0.0487)	$\beta = 55.5$ 0.6365 (0.0423)
n 30	ρ <sub>0</sub> 0.6204	eta = 75.5 0.5600 (0.0199)	eta = 80 0.5960 (0.0051)	eta = 80.5 0.6057 (0.0209)	eta = 85.5 0.6283 (0.0201)
n 40	ρ <sub>0</sub> 0.6136	eta = 110 0.6002 (0.0124)	eta = 110.5 0.6009 (0.0127)	eta = 111 0.6014 (0.0123)	eta = 111.5 0.6013 (0.0121)

**Table 4.** The simulated estimator  $\hat{\theta}_{\text{RDS}}$  and its variance for  $\theta = 2.5$  and for different values of  $\rho$  and *n*. Variances are given in parentheses.

n	ρ	$\theta_0$	$\hat{ heta}_{RDS}$
20	0.1	2.5252	2.4405 (0.3438)
	0.3	2.5216	2.4534 (0.3347)
	0.6	2.5140	2.4970 (0.3742)
30	0.1	2.5063	2.3841 (0.2141)
	0.3	2.5551	2.4140 (0.2073)
	0.6	2.5840	2.4293 (0.2189)
40	0.1	2.4949	2.3522 (0.1453)
	0.3	2.6455	2.3609 (0.1588)
	0.6	2.4982	2.3906 (0.1608)

We can see from all figures that the resulting estimate is not only very close to the true value of the parameter in each case, but it coincides with it in some cases (Figure 1 (c) and Figure 2 (c)).

#### 6. Conclusion

The proposed algorithm B2ARDS has allowed us to generate refined descriptive samples from input random variables for Bayesian estimation of AR(1) parameter in Monte Carlo simulation using the R software. As a consequence and apart from its already proved efficiency in the classical sense, we argue that the RDS method performs well even in a Bayesian context and allows the estimation of the AR(1) parameter using the Bayesian approach.

#### **Disclosure statement**

No potential conflict of interest was reported by the author(s).

# References

Albert, J. H., & Chib, S. (1993). Bayes inference via Gibbs sampling of autoregressive time series subject to Markov mean and variance shifts. *Journal of Business & Economic Statistics*, 11(1), 1–15.

Aloui, A., Zioui, A., Ourbih-Tari, M., & Alioui, A. (2015). A general purpose module using refined descriptive sampling for installation in simulation systems. *Computational Statistics*, 30(2), 477–490. https://doi.org/10.1007/s00180-014-0545-7

- Amaral Turkmann, M. A. (1990). Bayesian analysis of an autoregressive process with exponential white noise. *Statistics*, 21(4), 601–608. https://doi.org/10.1080/02331889008802270
- Andel, J. (1988). On AR(1) processes with exponential white noise. *Communications in Statistics Theory and Methods*, 17(5), 1481–1495. http://doi.org/10.1080/03610928808829693
- Ashby, D. (2006). Bayesian statistics in medicine: A 25 year review. Statistics in Medicine, 25(21), 3589-3631. https://doi.org/10.1002/sim.v25:21
- Baghdali-Ourbih, L., Ourbih-Tari, M., & Dahmani, A. (2017). Implementing refined descriptive sampling into three phase discrete-event simulation models. *Communications in Statistics : Simulation and Computation. Taylor and Francis*, 46(5), 4035–4049. https://doi.org/10.1080/03610918.2015.1085557
- Barnett, G., Koln, R., & Sheather, S. (1996). Bayesian estimation of an autoregressive model using Markov chain Monte Carlo. *Journal of Econometrics*, 74(2), 237–254. https://doi.org/10.1016/0304-4076(95)01744-5
- Beaumont, M. A., & Rannala, B. (2004). The Bayesian revolution in genetics. *Nature Reviews Genetics*, 5(4), 251–261. https://doi.org/10.1038/nrg1318
- Boubalou, M., Ourbih-Tari, M., Aloui, A., & Zioui, A. (2019). Comparing M/G/1 queue estimators in Monte Carlo simulation through the tested generator getRDS and the proposed getLHS using variance reduction. *Monte Carlo Methods and Applications*, 25(2), 177–186. https://doi.org/10.1515/mcma-2019-2033
- Box, G., Jinkins, G., & Reinsel, G. (1976). Time series analysis: Forecasting and control. John Wiley and Sons.
- Corander, J. (2006). Is there a real Bayesian revolution in pattern recognition for bioinformatics. *Current Bioinformatics*, 1(2), 161–165. https://doi.org/10.2174/157489306777011932
- Diaz, J., & Farah, J. (1981). Bayesian identification of autoregressive processes. In *Proceedings of the S.L, 22nd NBER-NSF.* Seminar on Bayesian inference in econometrics.
- Geweke, J. (1999). Using simulation methods for Bayesian econometric models. *Inference, Development, and Communication Econometric Reviews*, 18(1), 1–126. https://doi.org/10.1080/07474939908800428
- Ghosh, M., & Heo, J. (2003). Default Bayesian priors for regression models with first-order autoregressive residuals. *Journal of Time Series Analysis*, 24(3), 269–282. https://doi.org/10.1111/jtsa.2003.24.issue-3
- Ibazizen, M., & Fellag, H. (2003). Bayesian estimation of an AR(1) process with exponential white noise. *Journal of Theoretical and Applied Statistics*, *37*(5), 365–372. https://doi.org/10.1080/0233188031000078042
- Idjis, K., Ourbih-Tari, M., & Baghdali-Ourbih, L. (2017). Variance reduction in M/M/1 retrial queues using refined descriptive sampling. *Communications in Statistics – Simulation and Computation*, 46(6), 5002–5020. https://doi.org/10.1080/03610918. 2016.1140778
- Keller, A., Heinrich, S., & Niederreiter, H. (2008). Monte Carlo and quasi Monte Carlo methods. Springer.
- Kumar, J., & Agiwal, V. (2019). Bayesian estimation for fully shifted panel AR(1) time series model. *Thai Journal of Mathematics*, *17*, 167–183. https://thaijmath.in.cmu.ac.th
- Loh, W. L. (1996). On latin hypercube sampling. *Annals of Statistics*, 24(5), 2058–2080. https://doi.org/10.1214/aos/1069362310 Marriot, J., & Smith, A. (1992). Reparametrization aspects of numerical Bayesian methods for autoregressive moving average
- models. Journal of Time Series Analysis, 13(4), 307–331. https://doi.org/10.1111/j.1467-9892.1992.tb00111.x
- Mckay, M. D., Conover, W. J., & Beckman, R. J. (1979). A comparison of three methods for selecting values of input variables in the analysis of output from a computer code. *Technometrics*, 21(2), 239–245. https://doi.org/10.2307/1268522
- Monahan, J. (1984). Fully Bayesian analysis of ARMA time series models. *Journal of Econometrics*, 21(3), 307–331. https://doi.org/10.1016/0304-4076(83)90048-9
- Ourbih-Tari, M., & Azzal, M. (2017). Survival function estimation with non parametric adaptative refined descriptive sampling algorithm : A case study. *Communication in Statistics-Theory and Methods*, 46(12), 5840–5850. https://doi.org/10.1080/03610926.2015.1065328
- Ourbih-Tari, M., Zioui, A., & Aloui, A. (2017). Variance reduction in the simulation of a stationary M/G/1 queuing system using refined descriptive sampling. *Communication in Statistics-Simulation and Computation*, 46(5), 3504–3515. https://doi.org/10.1080/03610918.2015.1096374
- Pereira, I., & Amaral-Turkman, M. A. (2004). Bayesian prediction in threshold autoregressive models with exponential white noise. sociedad De Estadistica E Investigacion Operativa, 13(1), 45–64. https://doi.org/10.1007/BF02603000
- Saliby, E. (1990). Descriptive sampling: A better approach to Monte Carlo simulation. *Journal of the Operational Research Society*, 41(12), 1133–1142. https://doi.org/10.1057/jors.1990.180
- Smith, A. F. M., & Robert, G. O. (1993). Bayesian computation via the Gibbs sampler and related Markov chain Monte Carlo methods. *Journal of the Royal Statistical Society*, 5(5), 3–23. https://doi.org/10.1111/j.2517-6161.1993.tbo1466.X
- Stein, M. (1987). Large sample properties of simulations using latin hypercube sampling. *Technometrics*, 29(2), 143-151. https://doi.org/10.1080/00401706.1987.10488205
- Suparman, S., & Rusiman, M. S. (2018). Hierarchical Bayesian estimation for stationary autoregressive models using reversible jump MCMC algorithm. *International Journal of Engineering & Technology*, 7(430), 64–67. https://eprints.uthm.edu.my/id/ eprint/5110
- Tamiti, A., Ourbih-Tari, M., Aloui, A., & Idjis, K. (2018). The use of variance reduction, relative error and bias in testing the performance of M/G/1 retrial queues estimators in Monte Carlo simulation. *Monte Carlo Methods and Applications*, 24(3), 165–178. https://doi.org/10.1515/mcma-2018-0015
- Tari, M., & Dahmani, A. (2005a). Flowshop simulator using different sampling methods. *Operational Research*, 5(2), 261–272. https://doi.org/10.1007/BF02944312
- Tari, M., & Dahmani, A. (2005b). The three phase discrete event simulation using some sampling methods. *International Journal of Applied Mathematics and Statistics*, 3(5), 37–48.
- Tari, M., & Dahmani, A. (2006). Refined descriptive sampling: A better approach to Monte Carlo simulation. Simulation Modelling: Practice and Theory, 14(2), 143–160. https://doi.org/10.1016/j.simpat.2005.04.001

# **Appendices**

# Appendix 1 Computation of the $\rho$ values

According to formula (13), we have

$$\pi(\rho \mid x, \theta) \propto \rho^{\beta-1} e^{\theta \rho S}.$$
  
By applying the inversion method, we get  $F^{-1}(R_1) = \inf\{\rho : F(\rho) \ge R_1\}$  where  $R_1 \sim \mathcal{U}[0, 1].$   
We put  $R_1 = F(\rho) = \rho^{\beta-1} e^{\theta \rho S}$ , such that  $R_1 \sim \mathcal{U}[0, 1].$   
Then

$$\ln R_1 = \ln(\rho^{\beta - 1} e^{\theta \rho S})$$
$$= \ln(\rho^{\beta - 1}) + \ln(e^{\theta \rho S})$$
$$= \ln(\rho^{\beta - 1}) + \theta \rho S$$

and

where

$$\theta \rho S = \ln R_1 - \ln(\rho^{\beta-1}) = \ln\left(\frac{R_1}{\rho^{\beta-1}}\right),$$

$$\rho = F^{-1}(R_1) = \frac{1}{\theta S} \ln\left(\frac{R_1}{\rho^{\beta - 1}}\right)$$

# Appendix 2 Computation of the $\theta$ values

According to formula (14), we have

$$\pi(\theta \mid x, \rho) \propto \theta^{n-1} e^{-\theta(n\bar{x}-\rho S)}.$$
  
We put  $R_2 = H(\theta) = \theta^{n-1} e^{-\theta(n\bar{x}-\rho S)}$ , such that  $R_2 \sim \mathcal{U}[0,1]$ .  
Then

$$\ln R_2 = \ln(\theta^{n-1} e^{-\theta(n\overline{x} - \rho S)})$$
$$= \ln(\theta^{n-1}) + \ln(e^{-\theta(n\overline{x} - \rho S)})$$
$$= \ln(\theta^{n-1}) - \theta(n\overline{x} - \rho S)$$

and

$$\theta(n\overline{x}-\rho S) = \ln(\theta^{n-1}) - \ln R_2 = \ln\left(\frac{\theta^{n-1}}{R_2}\right),$$

where

so

 $\theta = \frac{1}{n\overline{x} - \rho S} \ln\left(\frac{\theta^{n-1}}{R_2}\right),\,$ 

$$\theta = H^{-1}(R_2) = \frac{1}{n\overline{x} - \rho S} \ln\left(\frac{\theta^{n-1}}{R_2}\right).$$

# Appendix 3 Computation of the MLE $\rho_0$ and $\theta_0$ by using the R language

We used the following programme to compute the MLE  $\rho_0$  and  $\theta_0$ .

Program >  $n = 30; \ \rho = 0.6; \ \theta = 2.5$ >  $x = arima.sim(list(ar = \rho), n, innov = rexp(n, rate = \theta))$ > mx = mean(x)> S = n \* mx - (x[n] - x[1])> for(*i* in 2 : *n*) s[i] = x[i]/x[i - 1]> s>  $\rho = min(1, s)$ >  $\rho$ >  $\theta = n/(n * mx - \rho * S)$ >  $\theta$