



HAL
open science

Robust tuning of Robbins-Monro algorithm for quantile estimation - Application to wind-farm asset management

Bertrand Iooss, Jérôme Lonchamp

► To cite this version:

Bertrand Iooss, Jérôme Lonchamp. Robust tuning of Robbins-Monro algorithm for quantile estimation - Application to wind-farm asset management. ESREL 2021, Sep 2021, Angers, France. hal-03191621

HAL Id: hal-03191621

<https://hal.science/hal-03191621v1>

Submitted on 7 Apr 2021

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.

Robust tuning of Robbins-Monro algorithm for quantile estimation - Application to wind-farm asset management

Bertrand Iooss & Jérôme Lonchamp

Department PRISME, EDF R&D, Chatou, France. E-mail: bertrand.iooss@edf.fr - jerome.lonchamp@edf.fr

In uncertainty quantification of numerical simulation model outputs, the classical approaches for quantile estimation requires the availability of the full sample of the studied variable. This approach is sometimes not suitable as large ensembles of simulation runs need to gather a prohibitively large amount of data and computer memory. This problem can be solved thanks to an on-the-fly (iterative) approach based on the Robbins-Monro algorithm. We numerically study this algorithm for estimating a discretized quantile function from samples of limited size (a few hundreds observations). We also define “robust” values of the algorithm parameters in two practical situations: when the final number of the model runs N is a priori fixed and when N is unknown in advance (it can then be minimized during the study in order to save cpu time cost). This method is applied to the estimation of indicators in the field of engineering asset management for offshore wind generation. We show how the proposed algorithm improves the efficiency of the tool to support risk informed decision making in the field of offshore wind generation.

Keywords: Uncertainty, Online statistics, Robbins-Monro, Averaging, Offshore wind, Operations and Maintenance.

1. Introduction

In engineering studies, a numerical simulation model is often used as a tool to assess the safety of a complex industrial system (de Rocquigny et al., 2008). For such a goal, when developing and using the numerical simulation model, uncertainty quantification (UQ) and global sensitivity analysis are valuable tools (Smith, 2014). It is then required to run the simulation model several (or even many) times with different values of the model inputs (according to their predefined probability laws) in order to calculate statistical quantities of interest (noted QoI) on the model outputs, i.e. their mean, variance, quantiles, probability of threshold exceedence, sensitivity indices, ... (Baudin et al., 2017). To estimate these QoI, the usual practice is to run all the simulation and store all their results before calculating the QoI.

A major difficulty arises when the UQ ensemble runs produce massive amount of data (e.g. when state and time variables are simulated) that have to be statistically aggregated, making them extremely vulnerable to the storage issues and I/O bottleneck. This is the case during the indicators' estimation phase in the field of engineering asset management for offshore wind generation. For large windfarms, asset-management Operations and Maintenance (O&M) models are assessed through Monte-Carlo simulation and risk-informed indicators such as quantiles are usually estimated a posteriori based on the results of all replications (Lonchamp et al., 2019). Saving these data leads to some issues regarding both computing time and memory that diminish the

efficiency of the tools to support decision making.

To keep a manageable amount of data, the classical approach, used in most of the UQ studies, consists in taking a limited number of outputs. In this paper, we adopt the more suitable technique based on one-pass statistical algorithms (also called iterative, recursive, update, online or even parallel statistics). Such algorithms only require to store the current results that can next be updated with incoming new samples: the required storage memory is therefore only the one needed for the results of one (or a limited number) simulation. This iterative statistical estimation issue is a relatively classical subject in the treatment of large volumes of data, in the so-called *big data* framework (Wang et al., 2016). However, iterative estimation issues have been little explored in UQ of numerical models (see Ribés et al. (2021) for an overview of the different underlying update statistical issues in UQ).

In this paper, we focus on the estimation of quantiles, as often required in simulation-based risk assessment, but in an iterative fashion. Indeed, quantiles are essential elements for the calculation of prediction or tolerance intervals, and for the detection of outliers, in particular in safety studies. As shown in Ribés et al. (2021), the Robbins-Monro based estimators of quantiles are robust and then good candidates to fix computing issues for wind-turbines O&M models. Indeed, the chosen algorithm has to be capable to deal with complex stochastic variables as the outputs may be multimodal, mixed discrete-continuous or supported on bounded or semi-infinite intervals. This paper shows how the proposed algorithm

improves the efficiency of the tool to support risk informed decision making in the field of offshore wind generation.

The following section presents the Robbins-Monro (RM) algorithm for quantile estimation. It allows to introduce the tuning RM parameters issue that is studied in this paper by the way of intensive numerical tests. Two different situations are then distinguished. First, the final number of iterations (i.e. number of computer model runs) N is a priori fixed, which is a classical way to deal with UQ problems. Section 3 defines well-tuned choices for the RM parameters in this context. Second, N is unknown in advance. Indeed, in practical situations, it is strongly interesting to stop the quantile estimation process when a sufficient precision has been reached in order to save cpu time cost. Section 4 shows the interest to use a stochastic adaptive stepsize rule associated to the averaged RM estimator in this context. Section 5 gives our application in the field of offshore wind generation and Section 6 concludes the work.

2. Quantile estimation

For the presentation of the algorithm, we consider a scalar output $Y \in \mathbb{R}$ which writes

$$Y = G(X) \quad (1)$$

where $G(\cdot)$ is the model function and $X \in \mathbb{R}^d$ is the vector of the d random input variables (defined by their joint probability density function). We look for an estimator \hat{q}_α of α -quantiles q_α (of the random variable Y) defined by:

$$q_\alpha = \inf\{y \in \mathbb{R} \mid \mathbb{P}(Y \leq y) \geq \alpha\}, \quad (2)$$

with $\alpha \in [\alpha_{\min}, \alpha_{\max}]$ where $\alpha_{\min} (\in]0, 1[)$ and $\alpha_{\max} (\in]0, 1[)$ are the minimal and maximal values of the orders of the estimated quantiles. In our study, α_{\min} (resp. α_{\max}) will be equal to 5% (resp. 95%). The empirical estimator of q_α writes

$$\hat{q}_\alpha^N = Y_{(\lfloor \alpha N \rfloor + 1)}, \quad (3)$$

where $(Y_{(1)}, \dots, Y_{(N)})$ is the ordered sample associated with the i.i.d. sample (Y_1, \dots, Y_N) . The sample (Y_1, \dots, Y_N) comes from a so-called Monte Carlo uncertainty propagation of N i.i.d. values of X through the model $G(\cdot)$ (see Eq. (1)).

Instead of this empirical estimator, the RM algorithm (Robbins and Monro, 1951) is devoted to iterative quantile estimation. It consists in updating the current quantile estimator (noted $q_\alpha(n)$) with each new observation Y_{n+1} with $n \geq 1$ by the recurrence formula

$$q_\alpha(n+1) = q_\alpha(n) - \frac{C}{n^\gamma} (\mathbb{1}_{Y_{n+1} \leq q_\alpha(n)} - \alpha), \quad (4)$$

with $q_\alpha(1) = Y_1$ (initialization step from the first data), $C > 0$ a constant and $\gamma \in]0, 1[$ governing

the rate of descent of the stochastic algorithm. The required memory storage for this estimator is two values (one realization of Y and one quantile estimate).

At finite sample size N , the RM estimator of the α -quantile of Y is therefore $\hat{q}_\alpha = q_\alpha(N)$. This estimator is consistent and asymptotically Gaussian for $\gamma \in]0.5, 1[$ (Duflo, 1997); this property provides essential guarantees of convergence. The value of γ does not therefore seem to be of crucial importance but, for low N , we will see that its adjustment is important. Indeed, in a lot of industrial studies, we have to deal with limited number of simulations (de Rocquigny et al., 2008) (typically a few hundreds) and, in this case, the asymptotic regime is not reached (Tierney, 1983).

3. Fixed number of model evaluations

In this section, the final size N of available realizations of the model output Y is fixed. The tuning of the γ parameter is first considered; then, the tuning of the C constant is studied.

3.1. Robust tuning of γ via a linear profile

We are looking for a value of γ which gives ‘‘acceptable’’ results whatever the distribution of Y (unknown in practice). Our numerical test considers the cases $Y \sim \mathcal{N}(0, 1)$ and $Y \sim \mathcal{U}[0, 1]$, with $N = 1000$, $C = 1$ and three orders of quantile α (0.05, 0.5 et 0.95). For each of these cases, Figures 1 and 2 show 50 independent trajectories of the RM estimator $q_\alpha(n)$ for $n = 1, \dots, N$ by considering three different choices of γ : 0.6, 1 and a linear variation as a function of n (Ribés et al., 2021) which is written

$$\gamma(n) = 0.5 + 0.5 \frac{n-1}{N-1}. \quad (5)$$

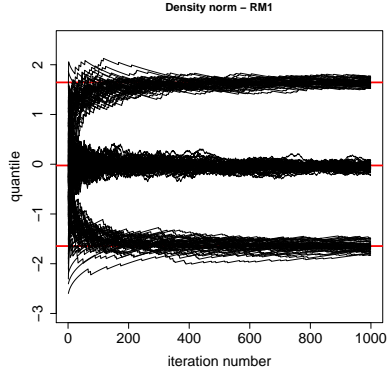
The theoretical and asymptotic properties of the RM algorithm are preserved using Eq. (5), because the γ values lie in $]0.5, 1[$.

The idea of the $\gamma(n)$ profile, given by Eq. (5), is to have strong fluctuations of the estimator at the beginning of the algorithm (to remove its dependence on the values of Y drawn first) then weak fluctuations at the end of the algorithm (to stabilize the estimator at the last iterations). This so-called *search-then-converge learning rule* (Powell, 2011) ensures a regular decrease of the mixing (from strong to weak) all along the RM iterations. Indeed, we can see on Figures 1 and 2 that the fluctuations with $\gamma = 1$ are too small in the Gaussian case ($\gamma = 0.6$ is satisfactory in this case) and the fluctuations with $\gamma = 0.6$ are too strong in the uniform case ($\gamma = 1$ is satisfactory in this case). The profile of a linear variation of γ achieves a compromise between these two extreme cases (and in many other tests carried out).

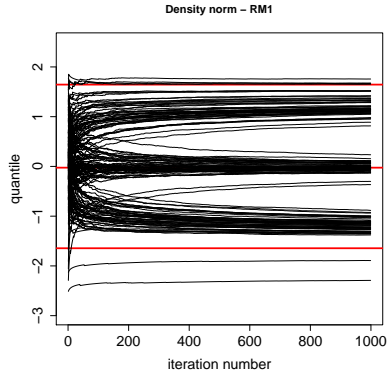
3.2. Robust tuning of C

In the previous section, the constant C has been set to 1. This choice turns out to be catastrophic when the variable considered has a dispersion which is not of this order of magnitude. It should

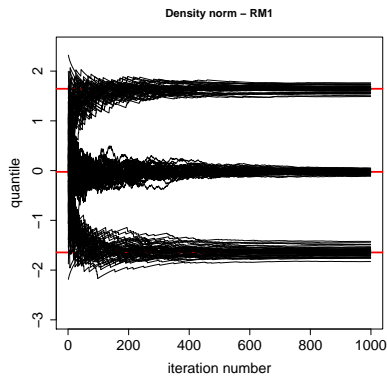
be remembered that in practice this dispersion of the studied variable Y is unknown. Then, as for γ , we look for a choice of C which gives “acceptable” results whatever the distribution of Y (unknown in practice).



(a) $\gamma = 0.6$.

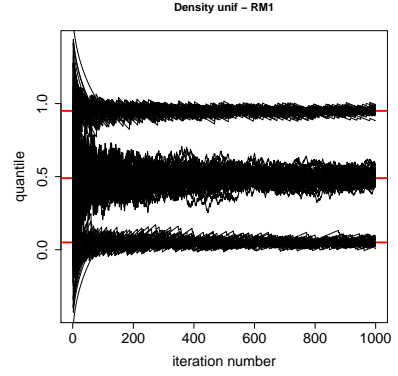


(b) $\gamma = 1$.

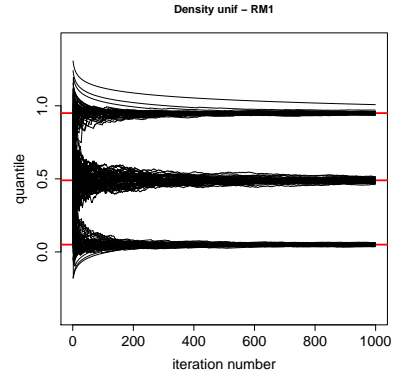


(c) Linear γ evolution.

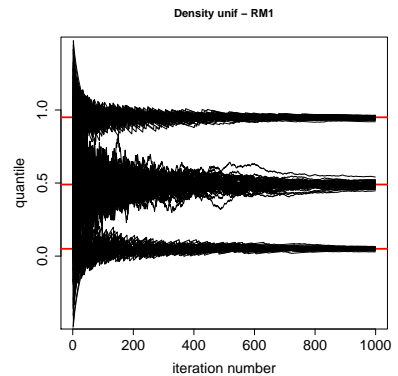
Fig. 1.: Simulations of trajectories of the RM algorithm ($N = 1000$, $Y \sim \mathcal{N}(0, 1)$). Red lines give exact order quantiles 0.05, 0.5 and 0.95.



(a) $\gamma = 0.6$.



(b) $\gamma = 1$.



(c) Linear γ evolution.

Fig. 2.: Simulations of trajectories of the RM algorithm ($N = 1000$, $Y \sim \mathcal{U}[0, 1]$). Red lines give exact order quantiles 0.05, 0.5 and 0.95.

Figure 3 shows 50 independent trajectories of the RM estimator $q_\alpha(n)$ for $n = 1, \dots, 1000$, Y following a lognormal law ($\log(Y) \sim \mathcal{N}(0, 1)$) and three orders of quantile α (0.05, 0.5 et 0.95). γ has a linear profile and different settings of C are tested: 1, 10 and an adaptive tuning which writes

$$C(n) = |q_{\alpha_{\max}}(n-1) - q_{\alpha_{\min}}(n-1)|, \quad (6)$$

where $n \geq 2$ and $C(1) = |Y_2 - Y_1|$. In Figure 3, it is clear that, for the quantile of order 0.95, C must be large enough to obtain sufficiently large fluctuations from the beginning of the RM algorithm. The adaptive fit of \bar{C} via Eq. (6) allows to automatically regulate these fluctuations. Other numerical tests on distributions of different types have confirmed the correctness of this choice.

4. Non-fixed number of model evaluations

From a user point of view, fixing the number N of model evaluations at the beginning at the study can be quite impractical. Indeed, the user has often no idea of the sample size that is needed to compute the QoI with a sufficient precision. Moreover, the profile that has been chosen in Section 3.1 for γ (Eq. (5)) depends on N . To get rid of the linear profile for γ , we first introduce an averaged RM version; then, a stochastic adaptive stepsize is proposed; finally, a simple stopping rule is given.

4.1. Averaged version of Robbins-Monro

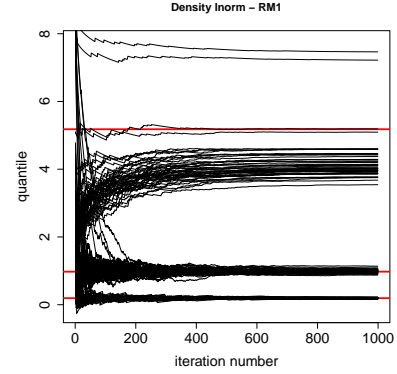
It is well known that the averaged version of RM (noted ARM) converges faster than the classical RM algorithm of Eq. (4) (Polyak and Juditsky, 1992). The idea is to exploit the recursive formula for computing a mean, to keep a quantile mean update (after applying the RM estimator) at each iteration. By preliminary numerical tests (not shown), we have however noted that, if an averaged quantile is introduced into (4), the fluctuations of the estimator along the iterations are not of sufficient magnitude to converge to the exact value. It is thus necessary to preserve the formulation (4) for $q_\alpha(n)$ and to store in addition, at each iteration, the averaged estimator (noted $\bar{q}_\alpha(n)$):

$$\bar{q}_\alpha(n+1) = \bar{q}_\alpha(n) + \frac{q_\alpha(n+1) - \bar{q}_\alpha(n)}{n+1}, \quad (7)$$

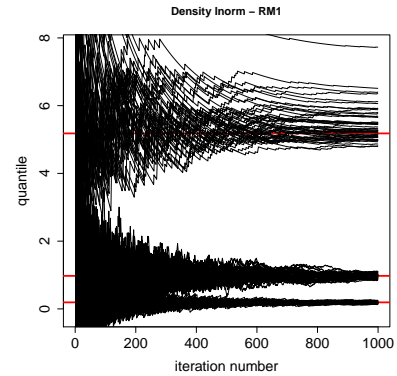
with $n \geq 1$ and $\bar{q}_\alpha(1) = Y_1$. The required memory storage for the ARM estimator is three values (one realization of Y and two quantile estimates).

Figure 4 compares the RM and ARM algorithms in the case $Y \sim \mathcal{N}(0, 1)$, $N = 1000$ and the adaptive tuning of C (Eq. (6)). The quantiles are estimated for orders α discretized inside the interval $[0.05, 0.95]$ by step of 0.01. The metric used (on the ordinate) is the mean square error between the exact quantiles and the estimated

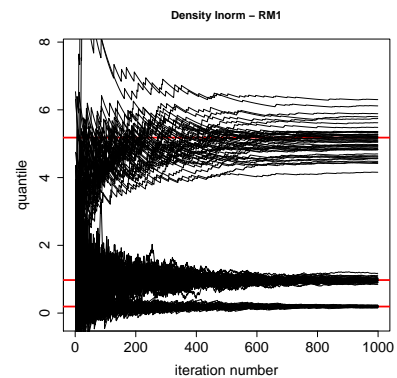
quantiles. Estimates are repeated 100 times independently in order to capture the variability of errors due to sampling. The reference estimator is the empirical estimator (which is not iterative). In these examples, the performances of ARM with



(a) $C = 1$.



(b) $C = 10$.



(c) Adaptive C .

Fig. 3.: Simulations of trajectories of the RM algorithm ($N = 1000$, $Y \sim \mathcal{LN}(0, 1)$). Red lines give exact order quantiles 0.05, 0.5 and 0.95.

a γ -linear profile ($\gamma(n)$) are similar and close to those of the empirical estimator, and better than those of RM. A constant and low γ (equals to 0.6) gives even better results with ARM (but not with RM) for both distribution cases (normal and uniform). In fact, the averaging in ARM (which makes the quantile estimator converges more quickly) makes it unnecessary to increase the γ towards 1 that we have with the linear profile.

Other tests with different distributions present similar conclusions. They also confirm that it is necessary to keep the adaptive tuning of C .

4.2. Application of the Kesten's rule

Another way of improvement of the RM algorithm would be to apply a stochastic adaptive stepsize rule (instead of a moving γ). The old and simple Kesten's rule (Kesten, 1958; Powell, 2011) allows to do so by incrementing n (in the stepsize $1/n$) only if the two last errors have different signs. It is based on the idea that "if we are far from the optimal, the errors tend to all have the same sign; as we get close, the errors tend to alternate". Then, it slows down the stepsize reduction at the beginning of the algorithm.

The RM estimator using the Kesten's rule (noted KRM) writes

$$q_\alpha(n+1) = q_\alpha(n) - \frac{C}{k_n^\gamma} (\mathbb{1}_{Y_{n+1} \leq q_\alpha(n)} - \alpha), \quad (8)$$

with $n \geq 1$, $q_\alpha(1) = Y_1$ and

$$\begin{aligned} \text{if } n > 2, \quad k_n &= k_{n-1} + \mathbb{1}_{\delta(n)\delta(n-1) < 0}, \\ \text{if } n \leq 2, \quad k_n &= n, \end{aligned} \quad (9)$$

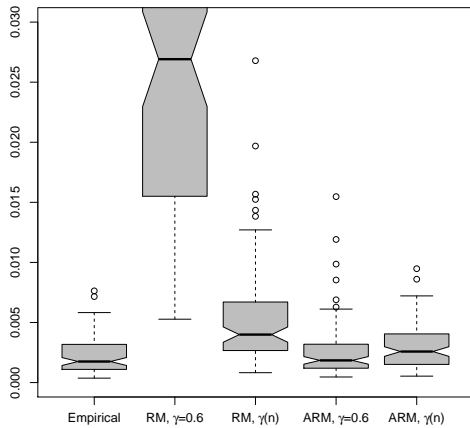
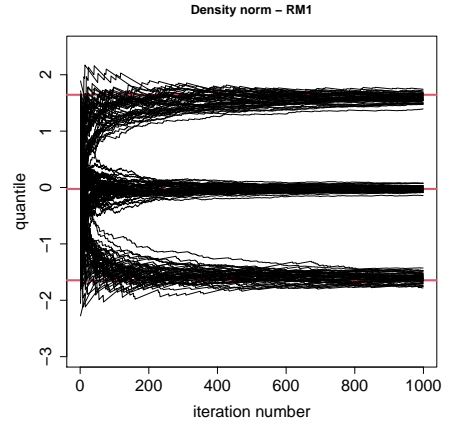


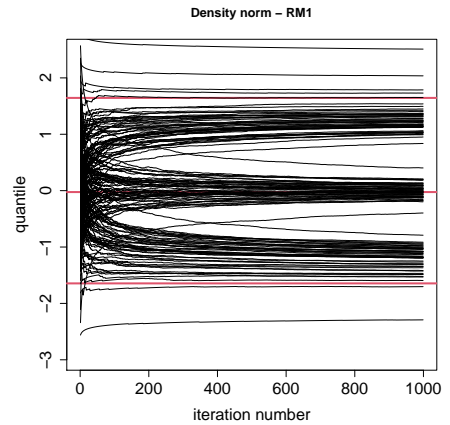
Fig. 4.: Mean square errors of discrete quantile functions for the empirical, RM and ARM estimators ($Y \sim \mathcal{N}(0, 1)$, $N = 1000$).

with $\delta(n) = q_\alpha(n) - q_\alpha(n-1)$. The required memory storage for the KRM estimator is four values (one realization of Y and three quantile estimates). If ARM (see section 4.1) is used instead of RM, the estimator is noted KARM and requires a memory storage of five values (one realization of Y and four quantile estimates).

Our numerical test considers the case $Y \sim \mathcal{N}(0, 1)$, with $N = 1000$, $\gamma = 1$, $C = 1$ and three orders of quantile α (0.05, 0.5 et 0.95). Figure 5 shows 50 independent trajectories of the RM estimator $q_\alpha(n)$ for $n = 1, \dots, N$ by applying Kesten's rule or not. Results clearly show that the Kesten's rule applies the *search-then-converge learning rule* (Powell, 2011) which led us to consider a γ -linear profile in Section 3.1. In this case, the KARM estimates provide similar result than the KRM (results not shown here).



(a) With Kesten's rule.



(b) Without Kesten's rule.

Fig. 5.: Simulations of trajectories of the RM algorithm ($N = 1000$, $Y \sim \mathcal{N}(0, 1)$). Red lines give exact order quantiles 0.05, 0.5 and 0.95.

Other tests with different distributions for Y (uniform, triangular, exponential, lognormal, multimodal, etc.) have been performed. The main conclusion is that KARM (Kesten's rule applied on the averaged RM estimator), associated to $\gamma = 1$ and the adaptive tuning of C , is the most robust approach.

4.3. Stopping rule

The algorithm has to be stopped by an automatic (and robust) rule. We look at the differences between two successive quantile estimates (using KARM) during a predefined number l_0 (≥ 1) of iterations. If N_t is the limit budget, for $l = 1, \dots, l_0$ with $2 + l_0 \leq n \leq N_t$, the stopping rule is:

$$|\bar{q}_\alpha(n-l+1) - \bar{q}_\alpha(n-l)| < \epsilon \forall \alpha. \quad (10)$$

If it is not fulfilled at $n = N_t$, algorithm stops.

5. Application to engineering asset management for offshore wind farm

Asset management processes, focused on realizing value from physical assets, have been developed for years. For the last one or two decades, these methods have been going from qualitative or semi-qualitative ones to quantitative management methods to support decision making. The generic framework is to build a model that describes both the technical and economic systems. For offshore wind generation, the main sources of uncertainty are failure dates and weather conditions that may impact access to the turbines (vessel transits) and working conditions on site. In order to help engineers optimize the hyperparameters of their O&M strategy, EDF has developed a tool that provides risk informed evaluations through Monte-Carlo simulation (Lonchamp et al., 2019). The approximated cumulative distribution functions (cdf) are built according to samples generated through out the simulation. However, the large number of data combined with the samples sizes needed for good convergence may lead to both memory (RAM memory for calculation and storage memory for results) and computing time (especially writing the results files) issues. In this section, we study the possibility for the RM algorithm to be used in this context, to provide quantile functions and improve the computing performance of the code.

The main indicators the engineers are interested in are the durations of maintenance tasks (stochastic due to weather conditions simulated with a probabilistic artificial neural network), as well as the availabilities of the wind turbines with an uncertainty associated to the failures dates and the task durations. Depending on the type of data, the stochastic variable will have different features regarding its bounds, its modes or its support continuity. Among all indicators, we focus on five typical ones with different features listed in Table 1, with examples of cdf given in Figures 6 and 7.

Two settings of the RM algorithm described above are implemented in the tool:

- The standard algorithm with a γ linear profile and an adaptive C parameter (RM), well suited to this application as the number of replications in the simulation is set by the user and therefore known a priori;
- The averaged version of the algorithm with the same dynamic parameters and the Kesten's rule (KARM).

All calculations are performed to evaluate all quantiles with a step of 0.01. Simulations are done on 1000 replications, leading to very large streamed samples (from 2×10^4 to 10^6 depending on the variable). The global performance of the algorithm is measured with the L^2 -distance to the empirical quantile function (which corresponds to the Wasserstein distance) obtained with entire samples with the same sizes as the ones used in Robbins-Monro calculations (F being the empiri-

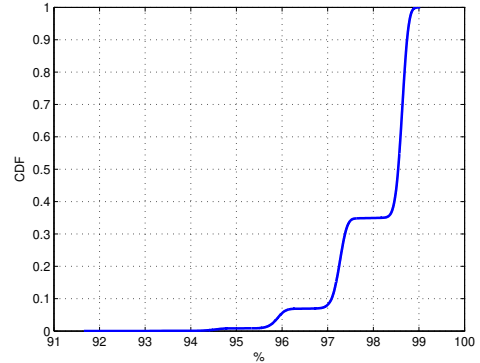


Fig. 6.: Cumulative distribution function of TOA.

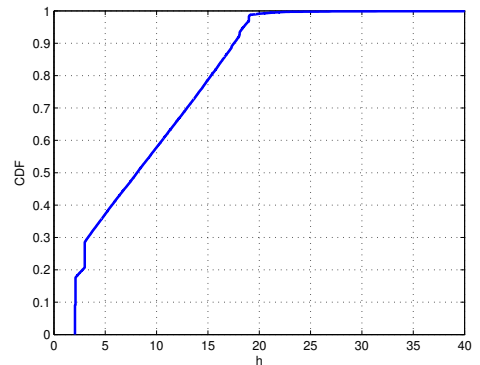


Fig. 7.: Cumulative distribution function of SRD.

Table 1.: List of variables and types (“s-i” means “semi-infinite”).

Variable	Support	Modes	Residual Risk
Farm annual availability (FAA)	bounded continuous	overlapping modes	no
Turbines overall availability (TOA)	bounded continuous	disjoint modes	no
Farm overall availability (FOA)	bounded continuous	unimodal	yes
Major repair duration (MRD)	s-i continuous	overlapping multimodes	no
Small repair duration (SRD)	s-i mixed continuous/discrete	overlapping multimodes	no

cal cdf):

$$W_2 = \left(\sum_{\alpha} (q_{\alpha} - F^{-1}(\alpha))^2 \right)^{1/2}. \quad (11)$$

The Figures 8 and 9 present examples of comparisons of the empirical evaluation of quantiles and the results obtained with the Robbins-Monro algorithms (RM and KARM).

From the results shown in Table 2, while the performance is globally good, it can be noticed that:

- RM and KARM algorithms are efficient for continuous and bounded variable, even multimodal if the modes are overlapping;
- Both algorithms have some difficulties when the quantiles are not well defined due to separated modes, as illustrated in Figure 8;
- Both algorithms have difficulties when evaluating the quantiles at singularities for mixed variables, especially for KARM (see Fig. 9);
- Globally there is little benefit in using the KARM algorithm over the simple RM one in this application. Mean values over 100 trials presented in 2 are lower with the RM implementation on all variables. The only benefit

coming from KARM is on its robustness with a range of W_2 from 1.5 to 3 time smaller.

Table 2.: Robbins-Monro algorithms performance on all variables as W_2 distance with empirical results, mean over 100 trials with minimal and maximal values.

Variable	RM	KARM
FAA	0.011 [0.006;0.0193]	0.012 [0.0102;0.0149]
TAA	0.034 [0.003;0.12]	0.04 [0.02;0.1]
TOA	0.052 [0.018;0.089]	0.054 [0.03;0.075]
SRD	0.08 [0.042;0.143]	0.102 [0.087;0.121]
MRD	0.09 [0.049;0.174]	0.111 [0.093;0.143]

If the performance of the RM seems good enough to make it an alternative to empirical quantification of risk indicators for offshore wind O&M, the goal to reach is the decrease in computing duration for simulation. Table 3 shows the computing times for both methods implemented

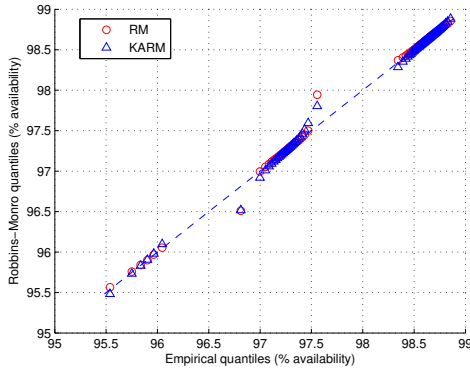


Fig. 8.: QQ plot of RM/KARM vs empirical results for TOA.

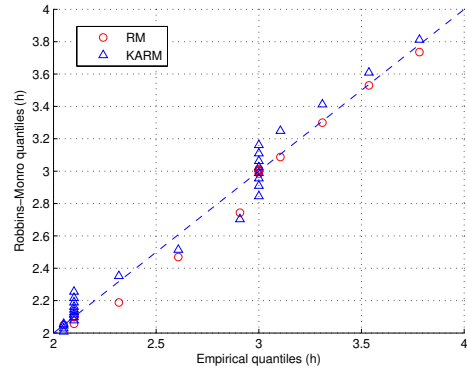


Fig. 9.: QQ plot of RM/KARM vs empirical results for SRD (zoom on small quantiles).

in the calculation engine on a real farm model with 50 turbines simulated on 1000 replications. These results demonstrate the important benefit given by RM methods as the total calculation time is reduced by over 35. Most of the differences come from the export function (2820 seconds vs 4 seconds with the RM method), with a very small increase of the simulation time due to the update process of the quantiles at each replication, even though it remains negligible compared with all the calculations done otherwise (4960 seconds in total for RM compared with 4860 for the current implementation).

Table 3.: Computing times comparison.

	Empirical estimate	RM estimate
Calculation (s.)	4860	4960
Export (s.)	2820	4

An additional computing benefit is the storage memory needed for a study. The largest models need up to 230 Mo with the current code when using the RM method brings the size of the output file down to 50 ko.

6. Conclusion

In order to avoid the storage of large amount of data during the stage of uncertainty propagation in computer models, this paper has proposed several robust versions of the RM algorithm. Some heuristics have been found for the iterative estimation of quantile of model outputs (Eq. (4)) with a finite-size sample N (a few hundreds values). Firstly, the choice of an adaptive C (Eq. (6)) is beneficial in all cases. Secondly, if N is known, the choice of a γ -linear profile (Eq. (5)) with RM or ARM (averaged version of RM) is robust and must be privileged. In this case, our applications has shown that RM methods give accurate enough performance regarding quantiles evaluation of key indicators for offshore wind engineering studies. Moreover, the important decrease in computing time makes it a promising alternative method that will be implemented in the next version of the code. Thirdly, the introduction of the stochastic adaptive stepsize Kesten's rule, which led to define the KARM estimator, gives good results (with $\gamma = 1$), while avoiding to define N at the beginning of the study. In such cases, a simple stopping rule is available.

These results need to be further investigated. For example, the use of well distributed point sequences instead of i.i.d. samples (tests not shown here) allow to greatly improve the precision of the

different RM estimators. It will also be fruitful to combine the RM algorithm and the techniques of simulation of rare events, as already studied by Kohler et al. (2014).

Acknowledgement

Part of this work has been funded by the international ANR project INDEX (ANR-18-CE91-0007) devoted to researches on incremental design of experiments.

References

- Baudin, M., A. Dutfoy, B. Iooss, and A. Popelin (2017). Open TURNS: An industrial software for uncertainty quantification in simulation. In R. Ghanem, D. Higdon, and H. Owhadi (Eds.), *Springer Handbook on Uncertainty Quantification*, pp. 2001–2038. Springer.
- de Rocquigny, E., N. Devictor, and S. Tarantola (Eds.) (2008). *Uncertainty in industrial practice*. Wiley.
- Dufflo, M. (1997). *Random iterative models*. Springer, Berlin.
- Kesten, H. (1958). Accelerated stochastic approximation. *Ann. Math. Stat.* 29, 41–59.
- Kohler, M., A. Krzyżak, and H. Walk (2014). Nonparametric recursive quantile estimation. *Statistics and Probability Letters* 93, 102–107.
- Lonchamp, J., A. Joly, T. Dawood, and A. Papatzimos (2019, june). An integrated asset management model for offshore wind turbines. In *Proceedings of the 29th International Ocean and Polar Engineering Conference*, Honolulu, Hawaii, USA. Springer.
- Polyak, B. and A. Juditsky (1992). Acceleration of stochastic approximation by averaging. *SIAM Journal on Control and Optimization* 30, 838–855.
- Powell, W. (2011). *Approximate dynamic programming, Second edition*. Wiley.
- Ribés, A., T. Terraz, Y. Fournier, B. Iooss, and B. Raffin (2021). The many advantages of avoiding writing files for large scale uncertainty quantification in numerical simulation. In H. Childs, J. Bennet, and C. Garth (Eds.), *In situ visualization for computational science*. Springer, In press.
- Robbins, H. and S. Monro (1951). A stochastic approximation method. *The Annals of Mathematical Statistics* 22, 400–407.
- Smith, R. (2014). *Uncertainty quantification*. SIAM.
- Tierney, L. (1983). A space-efficient recursive procedure for estimating a quantile of an unknown distribution. *SIAM Journal on Scientific and Statistical Computing* 4, 706–711.
- Wang, C., M.-H. Chen, E. Schifano, J. Wu, and J. Yan (2016). Statistical methods and computing for big data. *Stat. Interface* 9, 399–414.