Assessing the Efficiency of Variance Reduction Methods in the Construction Project Network Simulation

Sławomir Biruk¹, Piotr Jaskowski¹, Agata Czarnigowska¹

¹Faculty of Civil Engineering and Architecture, Lublin University of Technology, Nadbystrzycka 40, Lublin, 20-618, Poland.

s.biruk@pollub.pl

Abstract. The Monte Carlo simulation has become a standard tool in the practice of planning risk-affected projects. In particular, it is frequently applied to testing the impact of risk on schedule networks with deterministic structures and random activity durations defined by distribution functions of any type. The accuracy of simulation-based estimates can be improved by increasing the number of replications or by applying variance reduction methods. This paper focuses on the latter and analyzes the impact of the variance reduction method on the scale of the standard error of the estimated mean value of project duration. Three methods of variance reduction were examined: the Quasi-Monte Carlo with Weyl sequence sampling, the antithetic variates, and the Latin Hypercube Sampling. The object of the simulation experiment was a sample network model with the activity durations of triangular distributions. This type of distribution was selected as it is often applied in the practice of construction scheduling to capture the variability of operating conditions in the absence of grounds for assuming other types of distribution. The results of the sample simulation provided an indirect proof that applying variance reduction measures may reduce the time of the simulation experiment (reduced number of replications) as well as improve the confidence in the estimates of the model’s characteristics.

1. Introduction
The construction process is affected by risk; this justifies an assumption that durations of construction processes are random variables. The classic method of analysis applied to construction projects is PERT (Program Evaluation and Review Technique) developed in 1958. The popularity of the method arises from the fact that it greatly reduces the computational effort of the analysis network model under non-deterministic conditions, however, at the expense of reliability of the results. PERT assumes that the parameters of random numbers representing process durations can be credibly estimated on the basis of the experience of construction project practitioners – able to produce three figures for “most likely”, “optimistic” and “pessimistic” duration of each process. Further, PERT assumes that the total project duration is a normally distributed random variable, being a sum of independent random variables of the performance durations of processes on the critical path, according to the central limit theorem [1].

The latter assumption of an (approximately) normal distribution of project duration might hold if the critical path comprises more than 30 processes; in practical cases, 20 or even 10 processes are
considered sufficient to justify it. However, the accuracy of the completion date estimate largely depends on the number of critical processes and the similarity of their distribution types.

From the point of the probability theory, the assumption on the distribution type of the project duration is true only if the random variables of process duration are independent and if any process of the project starts directly on completion of its one and only predecessor. Consequently, PERT ignores the effect of the project paths that merge with the critical one, or the effects of multiple critical paths, both of them occurring often in real project networks. Therefore, PERT assumes that the moment of occurrence of an event of the project network is determined only by the path leading to a node representing it that comprises processes of the highest expected durations, and the remaining paths are neglected. The actual impact of other paths can be significant, especially when the length of the paths is not much different from the longest one (length expressed in terms of mean durations) and the variances of the random variables of process durations on these “neglected” paths are higher than the time variances of the path included in the calculation.

The consequence of the PERT’s assumption that each process must start directly on completion of its predecessor is the method’s ignoring strategies of process initiation other than “finish to start”. For instance, PERT was not designed to analyze network models with processes not allowed to start before a fixed date and having non-zero floats. In practical cases, many processes’ start dates are likely to be negotiated and agreed with co-operators (e.g. plant delivery dates, subcontracted works). A prerequisite for a practical planning tool is that it allows for not only the sequence of processes but also fixed dates for the project milestones or individual processes.

However, PERT, with its fixed network structure and task durations modeled as random values of probability distribution types and parameters based on historical records, is only one of a number of techniques to capture the uncertainty of schedules. Some researchers, while maintaining the assumption of deterministic network structure, choose fuzzy numbers to express the uncertainty of process durations and resource availability. A recent example of this approach the work of Khalilzadeh et al. [2]. Roghanian et al [3] presented an application of a fuzzy critical chain scheduling to size buffers. However, in practice, the project network structure is rarely determined: the sequence of tasks may have a number of feasible options, some planned tasks may prove unnecessary, etc. In such cases, random network models are adopted, using the classical probability theory (e.g. recent application of GERT network by Tao [4] or fuzzy logic [5, 6].

If a problem is so complex that exact methods cannot deliver a solution, simulations are applied. The main advantage of simulation models is the lack of constraints on the structure and complexity level of the investigated system allowing the user to model highly complex real-life systems with a large share of random factors [7]. Simulation experiments on a project network with non-deterministic process durations may provide insight into the scale of impact of the risk factors on particular processes and the project as a whole, as well as into the effect of the variance of process duration on the duration of the whole project [8].

Analyses of project network models by means of Monte Carlo (MC) digital simulation are typically aimed at estimating the mean values, the variances or the distribution types of the times of events (especially the project completion date) based on the observations \( t_i \) in \( i = 1, 2, \ldots, n \) simulation runs. The values of \( t_i \) can be construed as the realizations of the random variable \( T \) of the project performance duration. In each replication \( i \), the durations of the processes in a network model \( l = 1, 2, \ldots, w \), where \( w \) is the number of activities of the network model, are generated according to the applied probability distribution and by application of mutually independent series of random numbers \( u_{i,l} \) within an interval of \((0, 1)\). Then, the project duration is calculated as in the Critical Path
Method (CPM). Unbiased estimators of the mean $\mu = E(T)$ and the variance $D$ of random variable $T$ of the project performance are [9, 10]:

$$\hat{\mu} = \frac{1}{n} \sum_{i=1}^{n} t_i, \quad (1)$$

$$D^2(\hat{\mu}) = \frac{\sigma^2}{n}, \quad (2)$$

with $\sigma^2$ being the variance of random variable $T$, i.e., $\sigma^2 = D^2(T)$. The unbiased variance estimator is expressed as follows:

$$\hat{\sigma}^2 = \frac{1}{n-1} \sum_{i=1}^{n} (t_i - \hat{\mu})^2, \quad (3)$$

In the MC simulation method, the estimation accuracy is determined by the length of the confidence interval. The confidence interval size is proportional to $\sigma / \sqrt{n}$. One of the basic ways of narrowing down the confidence interval (and reducing the variance of the mean’s estimator) is to increase the number of observations (i.e., the number of simulation runs). This, however, extends the time to complete the simulation experiment. A number of variance reduction methods are used to reduce the time of simulation tests and improve the accuracy of results. Burt and Garman [11] provided a review of the methods, while Avramidis and Wilson [12] discussed the feasibility of their joint application. A particular method of variance reduction consists in applying a sampling method other than random sampling from theoretical probability distributions [13].

2. Sampling Applied in the Variance Reduction Methods for Pert Network Analysis

Monte Carlo makes it possible to use quasi-random sequences with a uniform distribution within an interval $[0, 1)$ as the baselines for the generation of activity durations in a network model. These numbers tend to fill out the interval $(0, 1)$ with a better uniformity than random ones. The quasi-random numbers also minimize the difference between the number of occurrences during a simulation in any interval of $[c, d] \subset [a, b]$ and the number compliant with the probability of occurrence [14]. Quasi-random numbers are also called low discrepancy sequences.

The method of antithetic variates consists of running a simulation experiment in two stages as explained below [15]. In the first stage, each activity has a series of random numbers $u_{l,1}, u_{l,2}, \ldots, u_{l,w/2}$ ($l = 1, 2, \ldots, w$) applied with a uniform distribution on the interval $(0, 1)$. In the second stage, the series of complementary numbers $(1-u_{l,1}), (1-u_{l,2}), \ldots, (1-u_{l,w/2})$ are applied. If $\hat{T}^{(i)}$ and $\hat{T}^{(2)}$ are, respectively, the estimators of the parameter $\hat{T}$ (the mean project duration) in the first and second stage of the simulation, then $\hat{T} = \frac{1}{2} (\hat{T}^{(i)} + \hat{T}^{(2)})$ is the unbiased estimator of this parameter. The variance of the estimator can be calculated as follows:

$$D^2(\hat{T}) = D^2\left(\frac{1}{2} (\hat{T}^{(i)} + \hat{T}^{(2)})\right) = \frac{1}{4} \left(D^2(\hat{T}^{(i)}) + D^2(\hat{T}^{(2)})\right) + \frac{1}{2} \text{cov}(\hat{T}^{(i)}, \hat{T}^{(2)}), \quad (4)$$

An inverse sampling with a negative correlation between the results of subsequent experiments provides a lower variance than running the experiments isolated from one another.
The literature on simulations of discrete systems presents a particular method of stratified sampling similar to inverse sampling. The simulation experiment is conducted in two iterations. In the first iteration, series of random numbers $u_{i,1}, u_{i,2}, \ldots, u_{i,n/2}$ of a uniform distribution on the interval $(0, 1)$ are applied, independently for each activity. In the second iteration, series that meet the following condition are applied:

$$u_{i,j} = \begin{cases} u_{i,j} + 0.5 & \text{for } 0 \leq u_{i,j} < 0.5 \\ u_{i,j} - 0.5 & \text{for } 0.5 \leq u_{i,j} \leq 1 \end{cases}$$

(5)

A negative correlation is applied between the random variables determined from the series of numbers $u_{i,j}$ and $u_{i',j}, i = 1, 2, \ldots, n/2$; this reduces the variance of the mean value estimator.

If the network model being the object of simulation experiment comprises many processes, Latin Hypercube Sampling (LHS) can be applied as proposed in [16]. This procedure generates estimators of lower variance than those produced by means of the classic MC. To generate $s$ $(i = 1, 2, \ldots, s)$ of correlated replications (where the number of replications must be equal to the number of strata), the values of random numbers $U_{i,l}$ (for the determination of the random variables) are to be defined as follows [17]:

$$U_{i,l} = \pi_i(i) - 1 + U_{i,l}^*$$

(6)

where $\pi_1(\cdot), \pi_2(\cdot), \ldots, \pi_w(\cdot)$ are permutations of the numbers $\{1, 2, \ldots, s\}$, randomly sampled with the same probability from a set $s!$ of such permutations, $\pi_i(i)$ is the $i$th element (the number of a stratum) in $l$th permutation, and $\{U_{i,l}^*: l = 1, 2, \ldots, w; i = 1, 2, \ldots, s\}$ are random numbers independent from each other and independent from the permutations $\pi_1(\cdot), \pi_2(\cdot), \ldots, \pi_w(\cdot)$

3. Example: Comparing Effects of Variance Reduction Methods on a Sample Network Simulations

The network model of a construction project, being the object of analysis, is shown in Figure 1. The random variables of activity durations, listed in Table 1, were assumed to be of triangular distribution with the parameters based on the estimates for the pessimistic, the most likely, and the optimistic duration.

Figure 1. Project network (the example)
Table 1. Process durations and the sizes of the work gangs in the project (example)

| Process nodes | Process description         | Optimistic duration $t_a$, days | Most probable duration $t_m$, days | Pessimistic duration $t_b$, days | Number of workers in the gang |
|---------------|----------------------------|---------------------------------|-----------------------------------|---------------------------------|-----------------------------|
| 1-2           | Earthworks                 | 5                               | 6                                 | 8                               | 5                           |
| 2-3           | Structure                  | 20                              | 30                                | 45                              | 4                           |
| 3-4           | Roof frame                 | 8                               | 10                                | 15                              | 10                          |
| 4-5           | Roof cladding              | 5                               | 7                                 | 10                              | 3                           |
| 5-6           | Façade                     | 20                              | 25                                | 35                              | 10                          |
| 3-7           | Partitions                 | 15                              | 17                                | 22                              | 6                           |
| 7-8           | Inner plasterwork          | 5                               | 7                                 | 10                              | 6                           |
| 8-9           | Inner primer painting      | 6                               | 8                                 | 11                              | 5                           |
| 9-10          | Inner painting             | 5                               | 7                                 | 10                              | 4                           |
| 10-15         | Flooring                   | 4                               | 5                                 | 7                               | 3                           |
| 11-12         | Mechanical & plumbing      | 24                              | 26                                | 29                              | 8                           |
| 13-14         | Electrical                 | 22                              | 25                                | 32                              | 5                           |

The following additional constraints were put on the project: at any time during the works, the total number of workers cannot be greater than 12, and Process 5 (façade) cannot begin before day 53. The sizes of the work gangs performing the processes are shown in Table 1.

Five simulation tests, differing in the sampling approach, were designed to compare the effect of sampling on the variance of the estimated total duration of the project. Each test comprised a total of 1000 replications.

The simulations were programmed in a general purpose simulation language, GPSS World from Minuteman Software. The language has a block structure. Each block of the 'ADVANCE' type represents a process, and the links between the blocks represent sequential relationships between the processes. Each simulation run is automatically controlled by the main program.

The basic simulation test used the GPSS World’s built-in triangular distribution generators \( \text{TRIANGULAR}(RN, t_a, t_b, t_c) \), where \( RN \) was the ordinal number of the stream of random numbers of a uniform distribution on the interval \((0, 1)\), and \( t_a, t_b, \) and \( t_c \) were the triangular distribution parameters.

Streams of random numbers with a different ordinal number \( RN \) were applied to each random variable. The GPSS World’s built-in distribution generators, including the triangular distribution generator, did not allow a direct application of certain variance reduction methods, for instance, the streams of antithetic numbers. Therefore, in further simulation tests, the inverse transform sampling method was applied to generate a triangular distribution with the parameters \((t_a, t_b, t_c)\) [15]:

\[
F^{-1}(u|t_a, t_b, t_c) = \begin{cases} 
  t_a + \sqrt{u(t_c-t_a)(t_b-t_a)} & \text{for } 0 \leq u \leq \frac{t_c-t_a}{t_b-t_a} \\
  t_b - \sqrt{(1-u)(t_b-t_c)(t_b-t_a)} & \text{for } \frac{t_c-t_a}{t_b-t_a} < u 
\end{cases}
\]

where: \( u \) is a random number with a uniform distribution in the interval \((0, 1)\).

In the experiment with inverse transform sampling method, the first simulation run generated the values of the random variables \( T_i \) of the activity duration from the random numbers \( RAN(l) = RN(l)/1000 \), where \( RN(l) \) is GPSS generator of uniform numbers in the range 1, …, 999. In the second simulation run, \( RAN(l) \) was replaced with \((1-RN(l))/1000\). The same was done in the experiment with
stratified sampling (expression 5). The inverse CDF method (expression 7) was used in both these experiments.

A Weyl’s sequence, defined by Equation 8, was used in the subsequent simulation tests [18]:

\[
x_i = n \cdot \Theta - \left\lfloor n \cdot \Theta \right\rfloor; \ i = 1, 2, \ldots,
\]

\[i = \theta_i - \theta_0, \quad \theta_i = \left\lfloor \frac{i}{n} \right\rfloor \theta_0. \quad (8)\]

A variety of bases \(\Theta_i\) of the Weyl’s sequences were used to generate the random variables of the activity durations in the network model:

\[
\Theta_0 = \frac{\sqrt{2} - 1}{2}, \ \Theta_1 = \frac{\sqrt{3} - 1}{2}, \ \Theta_2 = \frac{\sqrt{2} - 1}{2}, \ldots
\]

\[\Theta_i = \frac{\sqrt{2} - 1}{2}, \ \Theta_1 = \frac{\sqrt{3} - 1}{2}, \ \Theta_2 = \frac{\sqrt{2} - 1}{2}, \ldots. \quad (9)\]

The LHS method was based on the replicated Latin Hypercube Sampling to estimate the mean variance. It was possible to calculate the mean value estimator variance by repeating the experiment with \(s\) correlated replications \(k\) times [19]:

\[
D^2(\bar{\mu}) = D^2\left(\frac{1}{k} \sum_{u=1}^{k} \bar{\mu}_u\right),
\]

\[D^2(\bar{\mu}) = D^2\left(\frac{1}{k} \sum_{u=1}^{k} \bar{\mu}_u\right), \quad (10)\]

with \(\bar{\mu}_u\) being the mean calculated from \(s\) replications completed in a simulation run \(u (u = 1, 2, \ldots, k)\). In this example, each simulation run had 5 replications, and the experiment comprised 200 iterations.

Table 2 summarizes the results. The tests proved that the default random number generators of GPSS were of high quality (the sampled numbers uniformly filled the variance interval). The application of low dispersion or stratified sampling did not provide the expected results. Antithetic variates and LHS were the methods with very good effects. However, the first of the two was simpler to implement.

| Simulation method                                      | Mean of duration, days | Variance of mean duration |
|--------------------------------------------------------|------------------------|---------------------------|
| GPSS triangular generators                             | 154.8823               | 0.047557                  |
| Inverse transform method                               | 154.8394               | 0.05227                   |
| Sampling based on Weyl’s sequences                     | 154.8394               | 0.052218                  |
| Antithetic variates method                             | 155.0475               | 0.001303                  |
| Latin Hypercube sampling                               | 154.97776              | 0.005407                  |
| Stratified sampling                                    | 155.0706               | 0.052457                  |

4. Conclusions

The Monte Carlo simulation is an effective tool for analyzing project networks built of processes of any type of duration distributions and requires no additional simplifying assumptions. Computer simulations enable the planner to formulate and verify hypotheses on distribution type and parameters of occurrence of scheduled events and on the project's duration. Monte Carlo simulation of the project network enables the user to plan the project completion time at certain levels of probability or assess the impact of activity modes to select the best options. It also facilitates analyzing resource constraints in terms of the number of resource units available or the availability changing over time. The method
facilitates analyzing both simple models including only the dependence relationships and models with complex constraints on resource availability or time windows (e.g. the railway policy). The quality of results is primarily determined by the sampling policy. Application of variance reduction methods can reduce the time of analysis by reducing the number of replications and improve the reliability of estimates. Regarding the PERT network, the best results were always provided by the antithetic variates method and Latin Hypercube Sampling. However, the antithetic variates method was easier to implement in the simulation programs. The effectiveness of a specific variance reduction method depends on the configuration of the network, the types and parameters of the process durations, as well as the skills and professional experience of the researcher.

Acknowledgment
This work was financed from the core funds for statutory R&D activities by the Polish Ministry of Science and Higher Education. S/63. The authors gratefully acknowledge the support.

References
[1] F. S. Hillier, and G. J. Lieberman, Introduction to operational research, 7th ed.; McGraw-Hill: New York, USA, 2001.
[2] M. Khalilzadeh, H. Shakeri, H. Gholami, and L. Amini, “A Heuristic Algorithm for Project Scheduling with Fuzzy Parameters,” Procedia Comput. Sci., vol. 121, pp. 63-71, 2017.
[3] E. Roghanian, M. Alipour, and M. Rezaei, “An improved fuzzy critical chain approach in order to face uncertainty in project scheduling,” International Journal of Construction Management, vol. 18, no. 2, pp.1-13, 2017.
[4] L. Tao, D. Wu, S. Liu, and J. H. Lambert, “Schedule risk analysis for new-product development: The GERT method extended by a characteristic function,” Reliab. Eng. Syst. Saf., vol. 167, pp. 464–473, 2017.
[5] E. Radziszewska-Zielina, G. Śladowksi, and M. Sibiela, “Planning the reconstruction of a historical building by using a fuzzy stochastic network,” Autom. Constr., vol. 84, pp. 242-257, 2017.
[6] N. Ibadov, and J. Kulejewski, “Construction projects planning using network model with the fuzzy decision node,” Int. J. Environ. Sci. Technol., 2019 (in press)
[7] E. Tesfaye, K. Girma, E. Berhan, and B. Beshah, “A simulation of project completion probability using different probability distribution functions,” Afro-European Conference for Industrial Advancement. Advances in Intelligent Systems and Computing; vol. 334., pp. 133–145, 2015.
[8] S. E. Elmaghraby, “On criticality and sensitivity in activity networks,” Eur. J. Oper. Res., vol. 127, pp. 220–238, 2000.
[9] C. A. Chung, Simulation modeling handbook: a practical approach, CRC Press, Boca Raton, USA, 2003.
[10] R. Robinson, Simulation: The Practice of model development and use, 1st ed.; John Wiley & Sons Ltd.: Chichester, England, 2004.
[11] J. M. Burt, M. B. Garman, “Monte Carlo techniques for stochastic network analysis,” INFOR, vol. 9, pp. 248–262, 1971.
[12] A. N. Avramidis, and J. R. Wilson, “Integrated variance reduction strategies for simulation,” Oper. Res., vol. 44, pp. 327–346, 1996.
[13] H. Yang, “Quasi random sampling for operations management,” Seoul J. Bus., vol. 12, pp. 53–72, 2006.
[14] J. E. Gentle, Random number generation and Monte Carlo Methods, Springer, New York, USA, 2003.
[15] A. M. Law, and W. D. Kelton, Simulation Modeling & Analysis, 2nd ed.; Mc-Graw Hill, Singapore, 1991.
[16] M. D. McKay, R. J. Beckman, and W. J. Conover, “A comparison of three methods for
selecting values of input variables in the analysis of output from a computer code,” Technometrics, vol. 21, pp. 239–245, 1979.

[17] A.B. Owen, “Latin Supercube Sampling for very high-dimensional simulations,” ACM Trans. Model. Comput. Simul., vol. 8, pp. 71–102, 1998.

[18] S. Tezuka, Uniform random numbers. Theory and practice, Springer Science+Business Media: New York, USA, 1995.

[19] M. Stein, “Large sample properties of simulations using Latin Hypercube Sampling,” Technometrics, vol. 29, pp. 143–151, 2012.