*Proceedings of the 2014 Winter Simulation Conference*
*A. Tolk, S. Y. Diallo, I. O. Ryzhov, L. Yilmaz, S. Buckley, and J. A. Miller, eds.*

# SIMULATION-BASED MULTIOBJECTIVE OPTIMIZATION OF BRIDGE CONSTRUCTION PROCESSES USING PARALLEL COMPUTING

Shide Salimi
Mohammed Mawlana
Amin Hammad

Concordia University
1455 De Maisonneuve Boulevard West
Montreal, QC H3G 1M8, CANADA

## ABSTRACT

Conventionally, efforts are made to optimize the performance of simulation models by examining several possible resource combinations. However, the number of possible resource assignments increases exponentially with the increase of the range of available resources. Many researchers combined Genetic Algorithms (GAs) and other optimization techniques with simulation models to reach the Pareto solutions. However, due to the large number of resources required in complex and large-scale construction projects, which results in a very large search space, and the limittion of the GA capability in fast convergence to the optimum results, parallel computing is required to reduce the computational time. This paper proposes the usage of Non-dominated Sorting Genetic Algorithm (NSGA-II) as the optimization engine integrated with Discrete Event Simulation (DES) to model the bridge construction processes. The parallel computing platform is applied to reduce the computation time necessary to deal with multiple objective functions and the large search space.

## 1 INTRODUCTION

Due to the large number of factors affecting the construction and rehabilitation processes of bridges, these processes are highly complex for decision makers especially in terms of minimizing the time and cost of the projects. They have to find the optimum strategy to complete the projects successfully on time and within the budget considering all other constraints. Generally, there is an inverse relationship between the cost and time of a project because whenever the duration of a project is shortened, the cost of the project (i.e. the direct cost of labor, equipment, etc.) will increase considerably. Hence, finding a proper trade-off between these two key elements using optimization methods has become a crucial issue for project managers (Feng, Liu, and Burns 2000). On the other hand, simulation models are more and more needed in order to model the uncertainties associated with these projects (Yang, Hsieh, and Kung 2012). Several simulation tools are developed based on CYCLic Operation Network (CYCLONE) introduced by Halpin including MicroCYCLONE, STROBOSCOPE, COOPS, INSIGHT, RESQUE, and COST, which are especially designed for modeling construction operations (Halpin 1977). Therefore, the integration of simulation models with optimization techniques leads to an advancement in the decision making process.

Genetic Algorithms (GAs) have been widely used in different research areas to mimic natural selection and genetic mechanisms in order to find the optimum solutions (Holland 1975). Finding a set of Pareto-optimal solutions becomes crucial due to the conflict between the objective functions in most real-world problems (Deb 2005). Therefore, new solution techniques are needed to solve multi-objective optimization problems. Various multi-objective evolutionary algorithms (MOEAs) have been addressed in research (Zitzler and Thiele 1999). One of the first MOEAs was the Non-dominated Sorting Genetic

Algorithm (NSGA) (Srinivas and Deb 1994). In spite of the strength of this method in solving multi-objective optimization problems, its computational complexity, lack of elitism, and the need for sharing parameters are the main problems of the NSGA. Therefore, a modified algorithm called NSGA-II was introduced a few years later to alleviate some of these problems (Deb et al. 2002). The new algorithm performs better and faster to find the non-dominated solutions by providing a better distribution of the population.

Due to the large number of resources required in complex and large scale construction projects, such as bridge construction processes which results in a very large search space, and the limitation of the GA ability in fast convergence to the optimum results in this type of problems, there is a need for parallel computing in order to reduce the computational time. The master-slave (or global) parallelization and coarse-grained parallel GAs are two main parallel paradigms usually used when solving an optimization problem in order to improve the performance of the solver mechanism. In the global parallel GA, one processor called the *manager processor* is used to manage the communications of the other *worker processors*. The initial population of the GA is generated by the manager processor, and then, the generated population of solutions is broken down into subpopulations distributed equally to all other processors. The fitness evaluation is done by the worker processors and the evaluated solutions are sent back to the manager processor. The next generation of solutions is generated by performing generation evolution by the manager processor. This procedure is repeated until the convergence criterion is met (Kandil and El-Rayes 2006). The coarse-grained parallel GA uses a number of processors to perform the multi-objective optimization simultaneously without any need for the manager processor to manage them.

The main objective of this study is to investigate the parallel performance of the combination of a simulation tool such as SimEvents (MathWorks 2013) with a multi-objective optimization algorithm in one platform to reach the optimum resource combinations of bridge construction processes. Another objective is to reduce the computational effort by performing sensitivity analysis to tune the optimization algorithm and to find the best number of Central Processing Units (CPUs) used in parallel.

## 2 RELATED RESEARCH

Simulation plays an important role in helping managers to have better understanding of the construction processes and to make appropriate decisions (Touran 1990). Thus, many researchers used simulation for different construction processes (Hajjar and AbouRizk 2002; Lee and Ibbs 2005; Lee et al. 2010). On the other hand, optimization techniques are also widely used in order to find the optimal solutions of their problems especially time-cost tradeoff optimization of construction processes (e.g., El-Rayes and Kandil 2005; Xiong and Kuang 2008; Chen, Yang, and Su 2009; and Orabi et al. 2010).

Conventionally, the performance of simulation models was optimized by examining all possible resource combinations to find the best resource utilization which results in the best value for the output of the simulation model. However, if there is more than one simulation model for the construction process based on different configurations of shared queues, the evaluation of all these available options based on different possible resource combinations is a very costly and time consuming procedure which requires powerful computers with large memory capacity. In order to overcome these difficulties, Cheng and Feng (2003) developed a new deterministic simulation/single objective GA-integrated mechanism to find the optimal resource combination leading to the best performance of the construction operations. Cheng, Feng, and Chen (2005) proposed a similar GA-based modeling mechanism to optimize the resource allocation process as well as the modeling scheme which is mainly focused on the queue distribution within the simulation model. AbouRizk and Shi (1994) developed a heuristic algorithm (HA) to guide the simulation model in its search for the optimal resource allocation to attain some particular construction simulation objectives. However, they showed that the HA can improve the performance of the simulation model, but the HA searches for the local optimum solutions. Therefore, Cheng, Feng, and Hsu (2006) combined the proposed HA algorithm with GA, named heuristic GAs (HGA), in order to take advantage of both algorithms and reach to the global optimum results. Hegazy and Kassab (2003) developed a simulation tool called Process V3 combined with a commercial GA-based tool (Evolver DLL Routines) within Microsoft Project platform to optimize resource allocation while minimizing unit cost of large-scale projects. Mawlana and Hammad (2013) modeled a precast box girder concrete bridge construction process using STROBOSCOPE and developed a module in C# in order to combine the simulation model with the fast messy GA (fmGA) to optimize the construction cost and duration. However, the interoperability issues between the simulation and optimization tools make the integration procedure difficult and time consuming.

Kandil and El-Rayes (2006) investigated the efficiency and effectiveness of resource utilization optimization in large-scale construction projects by using parallel multi-objective GA. In order to improve the efficiency and effectiveness of multi-objective optimization algorithms (MOAs), two parallel computing paradigms including global and coarse-grained parallelization approaches were integrated with MOAs by Kandil and El-Rayes (2006) and Kandil, El-Rayes, and El-Anwar (2010). Das (1996) and Bisset (1998) designed some adaptive parallel DES (PDES) systems to be used for a variety of problems. Kartam and Flood (2000) compared the impact of using entity oriented parallel algorithm, recursive neural network method, and the conventional activity oriented serial algorithm, in order to speed-up the simulation procedure of construction processes within a multiprocessor computing environment. Yang, Hsieh, and Kung (2012) combined multi-objective particle swarm optimization (MOPSO) algorithm with Monte Carlo simulation for the bridge maintenance planning and implemented the propose framework in a parallel computing platform in order to reduce the computational burden associated with the problem. Master-slave, island, and diffusion are the three parallel programming paradigms used in their research. Super-linear speedups are attained using island and diffusion paradigms. In addition, all three parallel paradigms improve the solution quality when the number of cores is increased; however, the island platform outperforms the other two from the solution quality point of view within restricted time.

Most of the above mentioned works are based on the integration of Mont-Carlo or Discrete Event Simulation (DES) with optimization algorithms, or investigating the parallel performance of the integrated Mont-Carlo simulation models with optimization techniques, separately. Therefore, there is a need to develop a more practical and faster mechanism which combines the DES model with the optimization algorithm in a parallel environment.

## 3 RESEARCH METHODOLOGY

In this study, DES is combined with a multi-objective optimization algorithm (NSGA-II) in a parallel environment. First, the simulation methodology is introduced followed by a brief explanation of the optimization procedure. Thereafter, the proposed integrated framework is presented. Then, the parallel computing paradigm is described.

### 3.1 Simulation Model Based on DES

From the construction point of view, the simulation model contains a number of blocks indicating the required resources and activities with their durations to perform different tasks. Mont-Carlo simulation and DES are two popular methods to simulate construction processes. The contrast of DES with continuous simulation in which the system is continuously tracked over time (Robinson 2004) makes this type of simulation much faster than the continuous simulation models. In DES entities wait in queues to perform activities. DES models are mainly categorized in two modes, namely deterministic and probabilistic modes, based on the definition of the activities' durations, which are fixed for the former, while the latter is built by assigning random distributions to the activities' durations to consider the uncertainty associated with the process. Thus, every time the probabilistic model is run, a different set of outputs will be obtained due to the distinct seed numbers used in the simulation. In order to assess the risk associated with the model, replications are performed. The concept of replications aims to run the simulation model for a large number of times (e.g., 1000 times); and therefore, each replication will have different performance outcomes (e.g., time and cost). After having done the number of replications required, the means of the simulation outcomes are calculated (Nelson et al. 2013). In this study, these two modes are investigated.

As mentioned in Section 1, there are several special purpose tools to implement the simulation of construction processes with different advantages, limitations, and capabilities. In this study, the simulation model is created using the SimEvents component of MATLAB *Simulink Library* which is a general purpose simulation tool. However, simulating construction processes using special purpose simulation tools is much easier to build and understand than using SimEvents. On the other hand, the combination of those tools with optimization tools is difficult and integration platform is needed. Another advantage of using MATLAB is its capability in parallel computing, which is not applicable in most special purpose simulation tools. This parallel computing results in reducing the processing time in comparison with conventional integration solutions.

### 3.2 Optimization Procedure

GAs, as a metaheuristic method, mimic the process of natural selection in order to find proper solutions of optimization problems based on the ideas and techniques from genetic and evolutionary theory laid by Charles Darwin (Mitchell 1998). Unlike standard GAs which use binary strings of a fixed length, real numbers are used as genes in real-valued GAs; thus, the solution of the optimization problem is represented as a real vector $\{x_1, x_2, x_3, \ldots, x_n\}$. In this research, the decision variables of the multi-objective optimization problem are the integer numbers of different resources, which are represented in the queues, required in the simulation model. Therefore, the number of queues within the simulation model that are going to be optimized specifies the length of the chromosomes, and the value of each gene represents the number of required resources. Figure 1 illustrates the structure of the chromosomes in the GA used in this study. There are two approaches in order to deal with this type of GAs. In the first approach, real numbers are mapped to the binary strings of fixed length, and then, standard binary-string GAs are used to solve the problem. In the second approach, which is used in this study, the GA's standard operators (i.e., crossover and mutation) should be modified to be used in the real-valued GAs (Iba and Noman 2012). Applying a normal distribution of changes to genes (Gaussian noise) is one of the most straightforward ways of operating the real-valued GA's mutation. In the Gaussian mutation, a random number generated from a Gaussian distribution centered on zero and a predefined standard deviation is added to each gene (real number). The new mutated gene is then replaced in the chromosome (Hinterding 1995). There are different types of the crossover operator for creating new individuals in real-valued GAs (Iba and Noman 2012). In this study, an arithmetic crossover is selected to produce new children. The arithmetic crossover linearly combines two parents' chromosomes to produce two new children. If $\lambda_1$, and $\lambda_2$ are random numbers generated during the crossover operation, two new children are created according to equations (1) and (2):

$$\text{Child}_1 = \lambda_1 \text{Parent}_1 + \lambda_2 \text{Parent}_2 \tag{1}$$

$$\text{Child}_2 = \lambda_1 \text{Parent}_2 + \lambda_2 \text{Parent}_1 \tag{2}$$

Where $\lambda_1 + \lambda_2 = 1$ and $0 \leq \lambda_1, \lambda_2 \leq 1$. This restriction is called convex combination. On the other hand, affine combination would arise if there are no restrictions on the $\lambda's$ (Venkataraman 2009). Figure 2 shows how this operation works for the real-valued GA. Finally, rounding up and down are applied in order to reach to the integer number of resources.

### 3.3 Proposed Simulation-Based Optimization Framework

Decision makers are usually concerned with both the modeling methodology and finding the most appropriate way of resource usage to complete a project successfully within the budget and time constraints. Therefore, the integration of simulation and optimization is very important factor in construction processes. Figure 3 depicts the integration between DES and NSGA-II GA. The optimization algorithm starts with creating the initial population of size $N$ in the first generation. Each member of the population, goes through the simulation model. In the probabilistic mode of the simulation model, $K$ replications are performed for each set of variables, and the mean value of the objective functions (i.e., total cost and duration of the project) are calculated based on the results obtained from $K$ replications. After calculating the fitness values of all members of the population, the selection, crossover, and mutation operations are performed on the entire population. This procedure is repeated for all the members of the population in all generations until the convergence criterion is met (i.e., the specified number of generations). The same process is applicable in the deterministic mode with the difference that there are no replications in this mode.

### 3.4 Parallel Computing Approach

Due to the huge number of calculations resulting from the search space, multiple objective functions, and large number of replications performed in the stochastic simulation model, global parallel GA is used in this research to decrease the computation time and to efficiently use the full capacity of the computer as shown in Figure 4. This parallel platform is implemented on one server machine with a large number of CPUs. As mentioned in Section 1, one of the CPUs works as manager processor to create the initial population of GA, and then, this population is divided into subpopulations which are distributed among the remaining workers. The evolution of the subpopulations is the only parallelized operation in this parallel paradigm due to the fact that there is no dependency of the fitness evaluation of each individual on the rest of the population (Cantú-Paz 1998). Therefore, the workers run the simulation model to calculate the value of the objective functions. The communications between workers are necessary only when the CPUs receive a fraction of the population and after calculating the fitness values of individuals (Cantú-Paz 1998). After gathering the fitness values of the solutions based on the simulation model, the manager worker accomplishes the remaining parts of the multi-objective GA (MGA).

## 4 CASE STUDY

In this research, the precast full-span concrete box girder construction method with launching gantry is used to simulate the construction process of concrete bridges. The simulation process starts by using the resources needed for commencing the first task which is the erection of the reinforcement and stressing ducts of the bottom slab and the webs of the full-span, and then, the inner mold is installed followed by placing the reinforcement and stressing ducts of the top slab by steel crews. After finishing the reinforcement work, the rebar cage is put into an outer mold and the casting is done by the casting crews. When the concrete cured and reached an acceptable strength, the inner mold is removed. Next, the first pre-stressing procedure is performed by the pre-stressing crews to make the full-span ready for transportation to the storage area where the full-span is completely cured and stored (the second stage of pre-stressing process). After that, the precast full-span girder is transported to the site by means of a trailer for erection. The trailer hauls to the position where the onsite crane unloads the precast span from the trailer and loads it to a trolley. The trailer, then, returns to the storage area to load another span. The girder is simultaneously delivered along the completed part of the bridge by the trolley to its launching location where launching gantry repositions to the location of new span. Afterwards, the full-span is lifted from the trolley by means of the gantry's lifting frames, and the trolley returns to be loaded again. The girder is moved forward to reach to its right position to be placed between two piers. Then, the permanent bearings are installed to undertake the load of the span which is transferred from the temporary bearings to the permanent bearings. In the next step, the launcher repositions to lift the next full-span (Mawlana and Hammad 2013). Figure 5 illustrates the developed simulation model of bridge construction using precast full-span launching method.

# 5 VALIDATION OF THE PROPOSED MODEL

The proposed model is validated by comparing the optimal solutions obtained from NSGA-II optimization algoritm with those obtained from fmGA which is another type of GAs (Mawlana and Hammad 2013). Also, the performance of the parallel computing is investigated by calculating the performance in computation time reduction by increasing the number of CPUs. Like the proposed MGA, fmGA works based on the main principles of the simple GA with some modifications to alliviate the shortcomings of the messy GA and the simple GA. The fmGA consists of two loops which are the inner and outer loops. Each outer loop, called an era, perfoms an inner loop. The optimization starts with the random generation of the initial population within the inner loop. Then, the optimal solutions are evalutaed based on their fitness values and some selection and filtering processes (i.e., cut and splice, mutation) are applied on these solutions to increase the probability of finding the better solutions in the next generation. Finally, the optimization procedure stops when the termination critera are reached (Goldberg et al. 1993). Figure 6 compares the Pareto solutions attained by the proposed multi-objective NSGA-II and by the fmGA. In order to compare under the same main parameters, the size of the population and the number of generations are set to 100 and 2000, respectively, for both algorithms. The two optimization algorithms show almost similar behaviour in terms of Pareto solutions as shown in Figure 6. *Hypervolume indicator* can be used to compare Pareto sets by mesuring the volume of the dominated points of each Pareto set (Zitzler and Thiele 1999). In this sudy, to make the comparison procedure simpler, an averaging method is used. The results obtained from the averaging comparison demonstrates a 10.07% decrement in terms of the average duration of the project in the solutions generated by fmGA in comparison with those generated by NSGA-II. On the other hand, there is 7.36% increase in the total cost of the project when the fmGA algorithm is used as the optimization engine.

# 6 SENSITIVITY ANALYSES

## 6.1 Effect of Number of CPUs

The performance of the parallel platform is compared with respect to the computation time. The time which is needed to reach to the final optimum results using the proposed framework is calculated for different numbers of CPUs. For this purpose, all tests took place on a Server/Intel Xeon CPU E5540 @ 2.53 GHz, 48 GB Random Access Memory (RAM), running Windows 2010 Dell computer with Professional operating system and MATLAB R2013b environment. The performance of the problem is measured using the number of CPUs set from 1 to 12. In order to omit the effect of the uncertainty of the stochastic distribution of activity durations, all the durations are assumed to be deterministic as given in Table 1. Table 2 illustrates the computation time needed to complete the simulation-based optimization and the speedup achieved by increasing the number of CPUs. The speedup is calculated as S(1, n) = $T_1/T_n$, where $n, T_1$, and $T_n$ are the number of the CPUs, computation time obtained by using one CPU, and parallel computation time obtained by using $n$ CPUs, respectively (Yang, Hsieh and Kung 2012). As shown in Table 2, using two and three CPUs results in superliner speedup which means using these numbers of CPUs reduces the computation time less than 1/2, and 1/3, respectively. By increasing the number of CPUs more than 3, the near linear speedup is achieved. For instance, using 12 CPUs decreases the execution time of the proposed method to almost 1/8 of that needed when using a single CPU. Also, there is less improvement in the computation time when the number of CPUs exceed seven. These results show that the proposed method better uses the computer capacity to reach to the optimal solutions by saving time. The parallel performance of the stochastic simulation-based optimization procedure is also investigated by assigning random distributions to the activities' durations. In this case, the number of CPUs is set from 4 to 12 with an interval of 2. The results are shown in Figure 7. The results indicate that there is an almost linear relationship between increasing the number of CPUs and reducing the computation time.

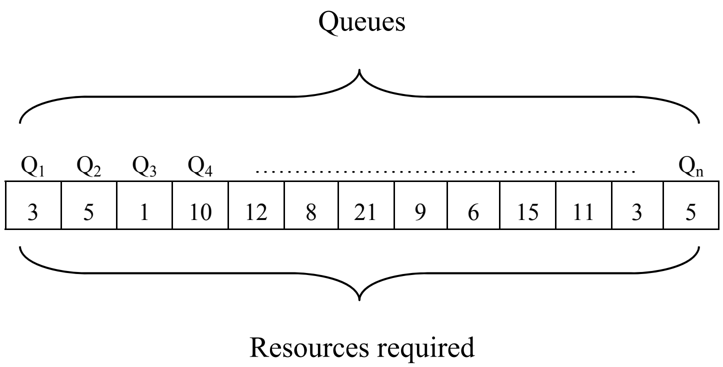


Figure 1: Chromosome structure of the real-valued GA.

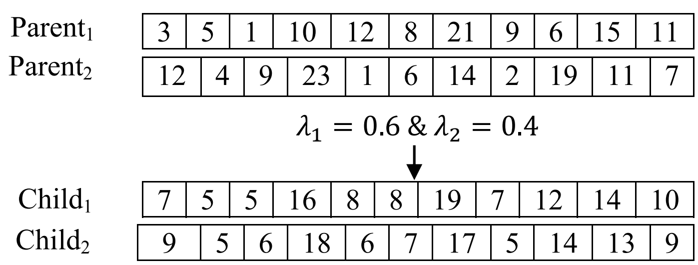


Figure 2: Arithmetic crossover operator for real-valued GA.

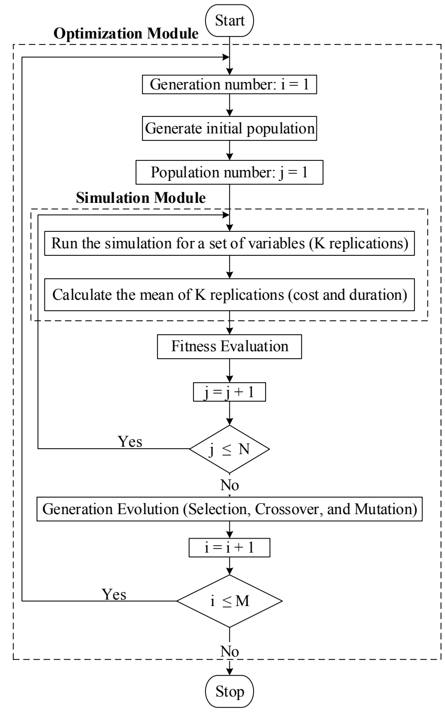


Figure 3: Parallel computing paradigm.

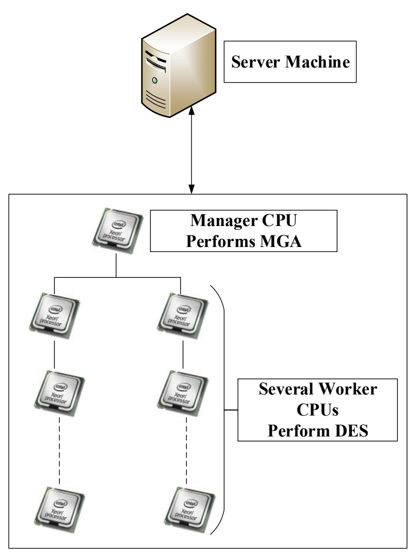


Figure 4: Integration of DES and NSGA-II.

## 6.2 Effect of GA Parameters

The main goal of applying the sensitivity analysis is to identify the most and less influential input parameters on the GA performance for the specific simulation-based optimization problem used in this paper. In order to achieve this goal, the effect of two main parameters of NSGA-II are investigated by varying one parameter while fixing the other.

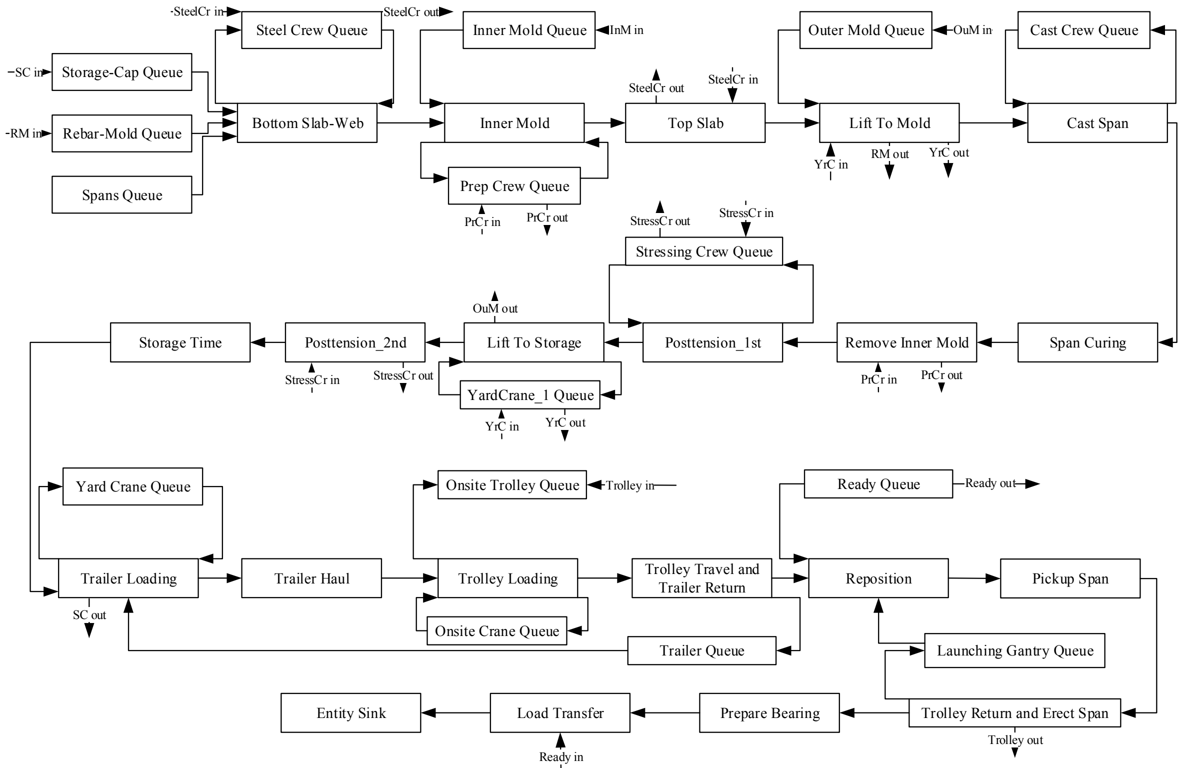



Figure 5: Simulation model of bridge construction using precast full-span launching method.

Table 1: Activity duration of deterministic simulation model (Mawlana and Hammad 2013).

| Activity | Duration (minutes) | Activity | Duration (minutes) |
|---|---|---|---|
| BottomSlab_Web | 1673 * | Trailer_Loading | 60 ** |
| Inner_Mold | 300 * | Trailer Haul | F (Distance, Speed) |
| TopSlab | 1979 * | Trolley_Loading | 60 ** |
| LiftToMold | 45 | Trailer_Return | F (Distance, Speed) |
| Cast_Span | 1544* | Trolley_Travel | F (Distance, Speed) |
| Span_Curing | (600 or 1200) * | Reposition | 240 ** |
| RemoveInnerMol | 255 * | Erection_Span | 240 ** |
| Posttension_1st | 240 * | Trolley_Return | F (Distance, Speed) |
| LiftToStorage | 60 ** | Prepare_Bearing | 240 ** |
| Posttension_2nd | 240 * | Load_Transfer | 60 ** |

* Adapted from (Marzouk, El-Dein and El-Said 2007)
** Adapted from (VSL International Ltd 2013)

Table 2: Parallel computation time.

| | Number of CPUs | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 |
| Time (min) | 1830 | 900 | 645 | 475 | 420 | 370 | 320 | 290 | 265 | 240 | 224 | 220 |
| **Speedup** | **1** | **2.03** | **2.84** | **3.85** | **4.36** | **4.95** | **5.72** | **6.31** | **6.91** | **7.63** | **8.17** | **8.32** |

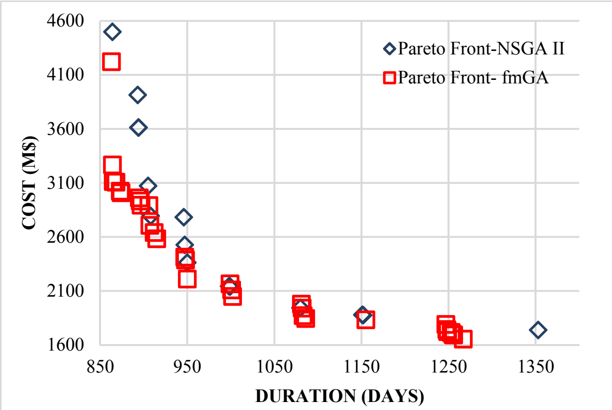



Figure 6: Comparison of Pareto solutions obtained from NSGA-II and fmGA.

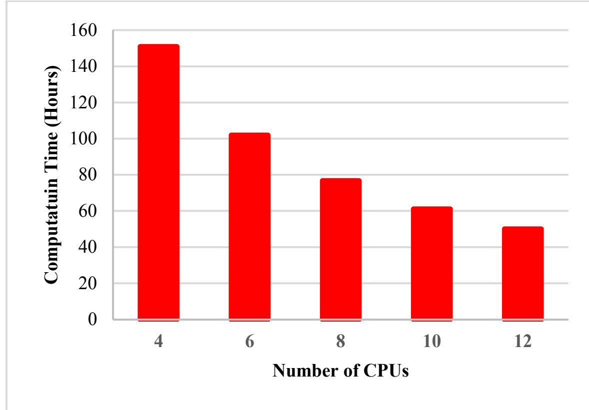


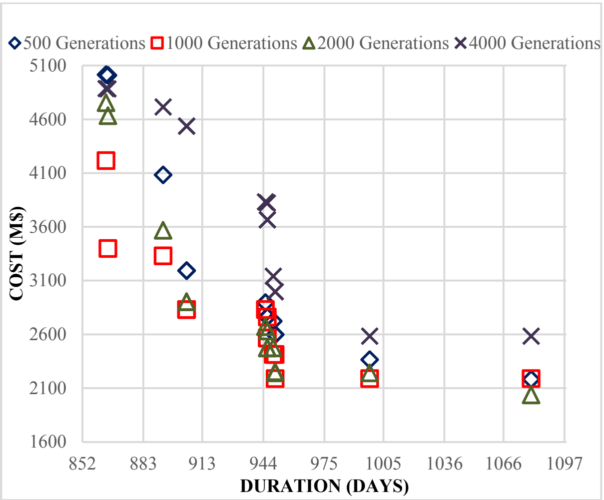

Figure 7: Saving in computation time by increasing the number of CPUs.

The population size and the number of generations are the two main parameters for checking the sensitivity of the output (i.e., quality of the objective functions) against the variations in their values. The GA is then tuned based on the best obtained parameters which maximize the GA's performance (Sugihara 1997). The larger population size results in more diversity of points in the search space which leads to better and more optimized results. On the other hand, when the size of the population increases, it takes more time and memory for the GA to converge. Practically, the population size of around 100 is very common for different purposes which can be changed based on the required balance between the time and memory of the computer and the quality of the final solutions (Sivanandam and Deepa 2008). In this study, the population sizes of 50, 100, 200, and 600 are selected while fixing the number of generations to 2000. The number of generations is another important factor when using the GA. In order to investigate the effect of this parameter on the values of the objective functions, this number is set to 500, 1000, 2000, and 4000. The sensitivity analysis is performed two times by fixing the population size to 100 and 200. The outcome of this analysis is expected to provide a better understanding of how changes in the various parameters affect the total cost and duration of the construction project.

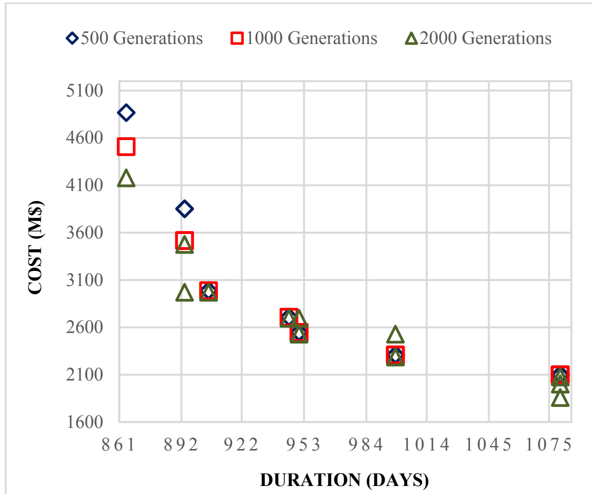
As shown in Figure 8, for the fixed population size of 100, the optimum solutions improve by increasing the number of generations until 1000 generations. More generations result in more costly solutions (e.g., the cost of the solutions for 4000 generations in Figure 8(a)); however, more diversity of solutions is observed when the number of generations is increased. The same behavior is seen for population of 200 individuals, except that the improvement in the optimum results continues with the increase of the number of generations. The proposed framework is also tested for various population sizes when the number of generations is fixed to 2000. The results are illustrated in Figure 9. As a rule of thumb, the bigger the population size, the better the optimum results that are likely to be reached; however, due to the extra computational burden imposed to the system as the population size increases, there should be a trade-off between the desired quality of the optimum solutions and the computation effort in terms of time and the memory capacity of system, which is determined based on experience and judgment. According to the closeness of the Pareto front solutions for different numbers of population size based on the results shown in Figure 9, the population size of 100 seems to be good from the solution quality and the diversity of the Pareto solutions in the case of the bridge construction problem.

## 7 CONCLUSIONS AND FUTURE WORK

A parallel simulation-based optimization framework is proposed in this study to model the concrete bridge construction process. In this framework, DES and multi-objective NSGA-II are integrated to find the optimum solutions based on the objective functions' values obtained from the simulation. The results show the good performance of the framework when compared to another GA (fmGA). Acceptable improvements in the computation time are achieved for both deterministic and stochastic simulation models using master-salve parallel paradigm. Furthermore, the NSGA-II algorithm is tuned based on the best parameters which maximize its performance by using sensitivity analysis. Due to the nondeterministic nature of NSGA-II, all results should be the mean values of several independent runs which is the next step in this research. The sensitivity analysis of other parameters of the GA, such as crossover and mutation probabilities, is considered as another future work. In addition, the computational power of the proposed framework can be improved by testing other parallel computing approaches.




(a)




(b)

Figure 8: Pareto solutions for different number of generations, with population size of 100 (a), and population size of 200 (b).

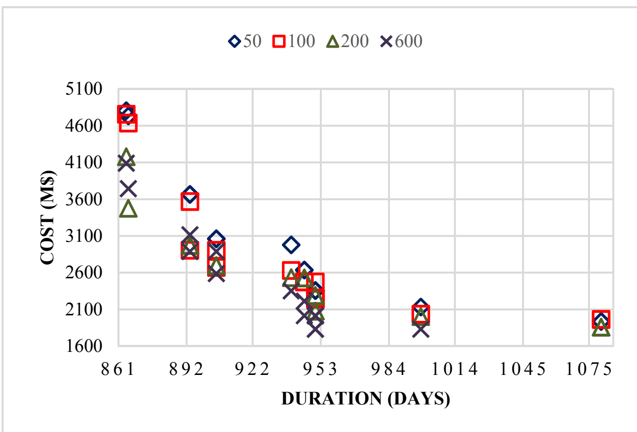



Figure 9: Pareto solutions for different population sizes with fixed 2000 generations.

## REFERENCES

AbouRizk, S., and J. Shi. 1994. "Automated construction-simulation optimization." *Journal of Construction Engineering and Management* 120 (2): 374-385.

Bisset, K. R. 1998. "An adaptive synchronization protocol for parallel discrete event simulation." *In Proceedings of the 31st IEEE Annual Simulation Symposium*, 26-33.

Cantú-Paz, E. 1998. "A survey of parallel genetic algorithms." *Calculateurs paralleles, reseaux et systems repartis* 10 (2): 141-171.
Chen, J. H., L. R. Yang, and M. C. Su. 2009. "Comparison of SOM-based optimization and particle swarm optimization for minimizing the construction time of a secant pile wall." *Automation in Construction* 18 (6): 844-848.
Cheng, T. M., and C. W. Feng. 2003. "An effective simulation mechanism for construction operations." *Automation in Construction* 12 (3): 227-244.
Cheng, T. M., C. W. Feng, and M. Y. Hsu. 2006. "An integrated modeling mechanism for optimizing the simulation model of the construction operation." *Automation in Construction* 15 (3): 327-340.
Cheng, T. M., C. W. Feng, and Y. L. Chen. 2005. "A hybrid mechanism for optimizing construction simulation models." *Automation in Construction* 14 (1): 85-98.
Das, S. R. 1996. "Adaptive protocols for parallel discrete event simulation." In *Proceedings of the 1996 Winter Simulation Conference*, edited by J. M. Charnes, D. J. Morrice, D. T. Brunner, and J. J. Swain, 186-193. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc.
Deb, K. 2005. "Multi-Objective Optimization." In *Search Methodologies Introductory Tutorials in Optimization and Decision Support Techniques*, edited by Edmund K. Burke and Graham Kendall, 97-125. United States of America: Springer Science-i-Business Media, LLC.
Deb, K., S. Agrawal, A. Pratap, and T. Meyarivan. 2002. "A fast and elitist multi-objective genetic algorithm: NSGA-II." *IEEE Transactions on Evolutionary Computation* 6(2): 182-197.
El-Rayes, K., and A. Kandil. 2005. "Time-cost-quality trade-off analysis for highway construction." *Journal of Construction Engineering and Management* 131 (4): 477-486.
Feng, C. W., L. Liu, and S. A. Burns. 2000. "Stochastic construction time-cost trade-off analysis." *Journal of Computing in Civil Engineering* 14 (2): 117-126.
Goldberg, D. E., K. Deb, H. Kargupta, and G. Harik. 1993. "Rapid, Accurate Optimization of Difficult Problems using Fast Messy Genetic Algorithms." In *Proceedings of the 5th International Conference on Genetic Algorithms*, 59-64. Urbana-Champaign, USA: Morgan Kaufmann Publishers Inc.
Hajjar, D., and S. M. AbouRizk. 2002. "Unified modeling methodology for construction simulation." *Journal of Construction Engineering and Management* 128 (2): 174-185.
Halpin, D. W. 1977. "CYCLONE- method for modeling job site processes." *Journal of the Construction Division* 103(CO3):489-499.
Halpin, D. W., and L.S. Riggs. 1992. *Planning and Analysis of Construction Operations.* Wiley.
Hegazy, T., and M. Kassab. 2003. "Resource optimization using combined simulation and genetic algorithms." *Journal of Construction Engineering and Management* 129 (6): 698-705.
Hinterding, R. 1995. "Gaussian mutation and self-adaption for numeric genetic algorithms." In *Proceedings of the 1995 IEEE International Conference on Evolutionary Computation*, 384-389. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc.
Holland, J. 1975. *Adaptation in Natural and Artificial System.* Ann Arbor, MI: Univ. of Michigan Press.
Iba, H., and N. Noman. 2012. *New Frontier in Evolutionary Algorithms.* London: Imperial College Press.
Kamil, K., N. M. M. Razali, and Y. Y. Teh. 2013. "Sensitivity analysis of GA parameters for ECED problem." In *Proceedings of the 7th IEEE International Power Engineering and Optimization Conference*, 256-260. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc.
Kandil, A., and K. El-Rayes. 2006. "Parallel Genetic Algorithms for Optimizing Resource Utilization in Large-Scale Construction Projects." *Journal of Construction Engineering and Management* 132 (5): 491-498.
Kandil, A., K. El-Rayes, and O. El-Anwar. 2010. "Optimization research: Enhancing the robustness of large-scale multiobjective optimization in construction." *Journal of Construction Engineering and Management* 136 (1): 17-25.
Kartam, N., and I. Flood. 2000. "Construction simulation using parallel computing environments." *Automation in Construction* 10 (1): 69-78.

Lee, D. E., C. Y. Yi, T. K. Lim, and D. Arditi. 2010. "Integrated Simulation System for Construction Operation and Project Scheduling." *Journal of Computing in Civil Engineering* 24 (6): 557-569.

Lee, E. B., and C. W. Ibbs. 2005. "Computer simulation model: Construction analysis for pavement rehabilitation strategies." *Journal of Construction Engineering and Management* 131 (4): 449-458.

Marzouk, M., H. Z. El-Dein, and M. El-Said. 2007. "Application of Computer Simulation to Construction of Incremental Launching Bridges." *Journal of Civil Engineering and Management* 13 (1): 27-36.

MathWorks. 2013. "Simevents R2013b user's guide." http://www.mathworks.com.

Mawlana, M., and A. Hammad. 2013. "Simulation-based Optimization of Precast Box Girder Concrete Bridge Construction Using Full Span Launching Gantry." In *Proceedings of the 4th Construction Specialty Conference,* Con-159-1 to Con-159-10.

Mitchell, M. 1998. *An introduction to genetic algorithms.* MIT press.

Nelson, B. L., D. M. Nicol, J. Banks, and J. S. Carson. 2013. *Discrete-Event System Simulation.* 5th edition. Pearson Education.

Orabi, W., A. B. Senouci, K. El-Rayes, and H. Al-Derham. 2010. "Optimizing Resource Utilization during the Recovery of Civil Infrastructure Systems." *J. of Manage. in Eng.* 26 (4): 237-246.

Robinson, S. 2004. *Simulation: The Practice of Model Development and Use.* Wiley.

Sivanandam, S. N., and S. N. Deepa. 2008. *Introduction to genetic algorithms.* Verlag Berlin Heidelberg: Springer.

Srinivas, N., and K. Deb. 1994. "Multi-objective Optimization using Non-dominated Sorting in Genetic Algorithm." *Evolutionary Computation* 2 (3): 221-248.

Sugihara, K. 1997. "Measures for performance evaluation of genetic algorithms." *In Proceedings of the 3rd Joint Conference on Information Sciences*, 172-175.

Touran, A. 1990. "Integration of Simulation with Expert Systems." *Journal of Construction Engineering and Management* 116 (3): 480-493.

Venkataraman, P. 2009. *Applied optimization with MATLAB programming.* Hoboken, New Jersey: John Wiley & Sons, Inc.

VSL International Ltd. 2013. *Bridge Construction Technology.* Accessed January 23, 2013. http://www.vsl.com.

Xiong, Y., and Y. Kuang. 2008. "Applying an Ant Colony Optimization Algorithm-Based multiobjective approach for time–cost trade-off." *J. of Const. Eng. & Manage.* 134 (2): 153-156.

Yang, I.T., Y.M. Hsieh, and L.O. Kung. 2012. "Parallel computing platform for multiobjective simulation optimization of bridge maintenance planning." *J. of Const. Eng. & Mange.* 138 (2): 215-226.

Zitzler, E., and L. Thiele. 1999. "Multiobjective evolutionary algorithms: a comparative case study and the strength Pareto approach". *IEEE Transactions on Evolutionary Computation* 3(4): 257-271.


## AUTHOR BIOGRAPHIES


**SHIDE SALIMI** is a Research Assistant and M.S. student in BCEE Department at Concordia University. Her research area is mainly focused on simulation-based optimization of construction processes using parallel computing. Her e-mail is sh_sa@encs.concordia.ca.

**MOHAMMED MAWLANA** is a Research Assistant and PhD student in BCEE Department at Concordia University. His research interests are automation in construction and simulation optimization. His e-mail is m_mawlan@encs.concordia.ca.

**AMIN HAMMAD** is a Professor at the Concordia Institute for Information Systems Engineering. His research focuses on sustainable infrastructure lifecycle management systems and investigates several methods and techniques including spatio-temporal information modeling and analysis, simulation, visualization, optimization, wireless communications, sensing, auto-identification, and real-time location tracking. His email is hammad@ciise.concordia.ca.