Proceedings of the 2018 Winter Simulation Conference
M. Rabe, A.A. Juan, N. Mustafee, A. Skoogh, S. Jain, and B. Johansson, eds.

MACHINE LEARNING BASED CONSTRUCTION SIMULATION AND OPTIMIZATION

Kailun Feng
Shiwei Chen

Department of Construction Management
Harbin Institute of Technology
Fayuan Street 13
Harbin, 150001, CHINA

Weizhuo Lu

Department of Civil, Environmental, and Natural Resources Engineering
Luleå University of Technology
Luleå, 97187, SWEDEN

ABSTRACT

Building construction comprises interaction and interdependence among processes. Discrete-event simulation (DES) is widely applied to model these processes interaction. To find optimal construction plans, optimization technique is usually integrated with DES. However, present simulation-optimization integrated method directly invokes simulation model within optimization algorithms, which is found significantly computationally expensive. This study proposes a machine learning based construction simulation and optimization integrated method. After trained by DES, the machine learning model accelerates simulation-optimization integration by nearly real-time providing fitness evaluation within optimization. This method was implemented into a real construction project for construction time-cost-environment optimization. Results show that proposed machine learning based method significantly reduce computing time compared with original simulation-optimization integration. Less than 1% of construction cost and time improvement were miss, while greenhouse gas emissions obtained same performance. The new method could be a more effective DES and optimization integration approach for practical engineering application.

1 INTRODUCTION

Building construction is often a complex process where the construction activities and resources have interactions with each other (Lu and Olofsson 2014; Segerstedt and Olofsson 2010). Simulation has been used as an effective tool to analyze the complex interactions and uncertainties during the construction process (Hajjar and Abourizk 2002). However, merely based on simulation, comparing every possible alternative and find the optimal construction plans are time-consuming and impractical.

In order to reduce the time of analysis, a simulation and multi-objective optimization integrated method (SO) was proposed (Inyim and Zhu 2013). In this way, the simulation plays a role as evaluation tool inside of optimization iterations. Thus, the optimization algorithms search in a great number of possible solutions, while the discrete-event simulation (DES) captures the complex relations of studied process system. This will avoid the assessment for all of alternatives and find near-best solutions. Several studies have applied this integrated approach in construction planning. Zhang and Li (2004) developed an optimization method combining DES and a heuristic algorithm to minimize overall construction duration. Li et al. (2017) used an integrated DES and genetic algorithm (GA) optimization approach to optimize labor allocation and reduce CO_2 emissions caused by on-site construction. Shin et al. (2011) proposed a DES model incorporating GA which could help contractors quickly find the optimal hoist plans for high-rise building construction

However, as many optimization procedures especially the meta-heuristics follow an evolutionary process of populations. In addition, discrete-event simulation requires level of replications to obtain robust results. Previous integration method, which is directly invoking of discrete-event simulation every time during optimization evaluation, can be significantly computationally expensive (Yousefi et al. 2018). It will cost too much time (usually days or weeks) to perform optimization when the simulation model is complex, which could make SO an impractical approach for construction engineering practices.

This study applied a Machine Learning (ML) based method to accelerate construction process simulation and optimization integration. Monostori and Viharos (2001) pointed out that artificial neural network (ANN) can substitute evaluation function of discrete-event simulation to save time of running simulation during simulation-based optimization in production field. Yadav et al. (2017) built an integrated simulation-optimization method based on ML to design an in-situ bioremediation system with minimal cost. Yousefi et al. (2018) used ML to connect agent-based simulation and GA to plan optimum resource allocation in emergency departments. However, in construction field, where discrete-event simulation are widely applied to analyze construction processes, the benefit of machine learning based simulation-optimization integration have not been fully understand. In order to fulfill this gap, this study proposes a Machine Learning based simulation and optimization (MSO) integration approach for construction planning. The ML model will extract the connections between the inputs and outputs of the simulation model. Then ML model can real-time provide the outputs based on inputs in optimization algorithms. Therefore, this novel integration method could reduce mass of simulation running time in optimization iteration. Proposed MSO method might be an industry practical tool for actual construction engineering problems.

This paper is structured as follows. In Section 2, the method combining simulation, multi-objective optimization and ML is proposed. In Section 3, a case study is conducted to compare both the optimization performance and speed between SO and MSO. In Section 4, some advantages and limits of MSO will be discussed based on the results given in Section 3, and a summary of the contributions of the proposed method will be presented.

2 METHODS

The overall machine learning based discrete-event simulation and multi-objective optimization integration follows the framework as Figure 1. The discrete-event simulation is built based on construction process logic. Machine learning is employed to build a pseudo evaluation model for construction planning, which can nearly real-time provide evaluation within multi-objective optimization algorithms. The testing procedure is conducted for learning model until its predicted performance is accepted. After that, the optimization searching iteration is performed based on established learning model, until pseudo Pareto solutions are obtained. Then, these pseudo solutions are re-simulated by simulation model to get actual simulation results. After getting rid of non-Pareto solutions, the optimal construction operation is obtained.

2.1 Discrete-Event Simulation

Construction operation processes usually have interaction and interdependence among each other (Lu and Olofsson 2014). Changing of one construction process, for instance, re-configuration of equipment, construction crew, etc., may propagate to others and entire system. DES has the ability to capture interaction and interdependence among complex construction processes (Larsson et al. 2015). In this integration framework, DES is designed to simulate construction operation processes and provides process data for construction cost, duration, and GHG emissions assessment. To precisely model the characteristics of the construction process, a construction production model referred to as *Activity-Component-Resource-Action-Sequence* (CARS) (Fischer et al. 1999), is used to establish construction DES model. The *Activity* is conducted construction task, and *Component* is the constructed building component, such as building beam, column, and slab at specific building location. *Resource* is defined as construction equipment and labor resources. *Action* is executed work information for each activity, and *Sequence* is the logic restriction of the construction process. "CARS" collects basic information for a construction DES model.

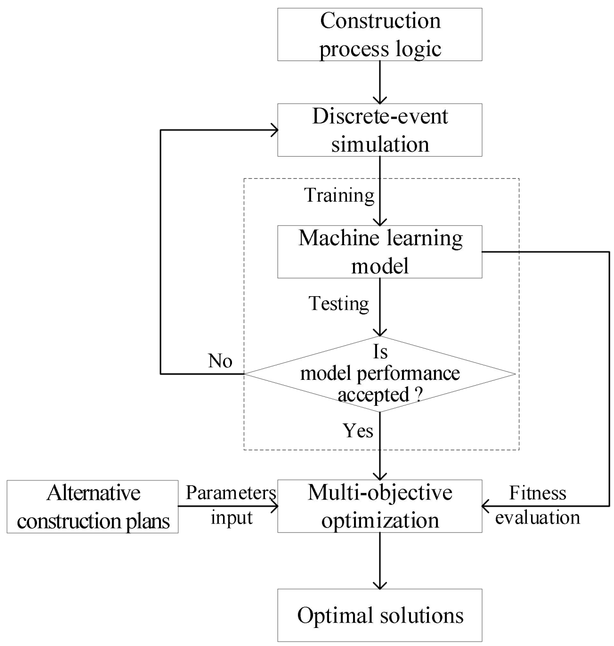

Figure 1: Integration framework.

The DES model simulates construction process data including quantity of work (*Q*), distributed working productivity (*WP*), working time (*WT*), and idle time (*IT*) of each equipment and construction crews. These process data will be used to assess construction cost (*C*), and GHG emissions (*E*) as Eqs (1) and (2). Where c_e and c_w are cost of equipment and labor unit working time, respectively; *e* is GHG emissions unit equipment working time. As discrete-event simulation has the ability to capture the interaction and interdependence among construction processes, it is defined to directly simulation total construction duration. The simulation should be set to have replications to obtain reliable results. A method proposed by Law and Kelton (2007) tests the coefficient of variation (*Cv*) of outputs with increased number of replication. It obtains the minimum number of replication when outputs get stable.

$$C = c_e \times (Q/WP + IT) + c_w \times Q/WP. \tag{1}$$

$$E = e \times WT. \tag{2}$$

2.2 Machine Learning for Integration

This study proposes a novel simulation-optimization integration method to address the computation load. Machine learning model, artificial neural network (ANN) in this study, is constructed to learn knowledge of construction discrete-event simulation. This learning model is defined to extract the instinct relations between construction configuration and outcomes of discrete-event simulation (see Figure 2). After validation of the learning model, it replaces discrete-event simulation to be called for fitness evaluation within optimization. The number of simulation samples and size of hidden layer are decided based on case complexity. And all the samples for ANN are divided into training and testing sets with 70% and 30%. The Bayesian regulation backpropagation is applied to train ANN model.

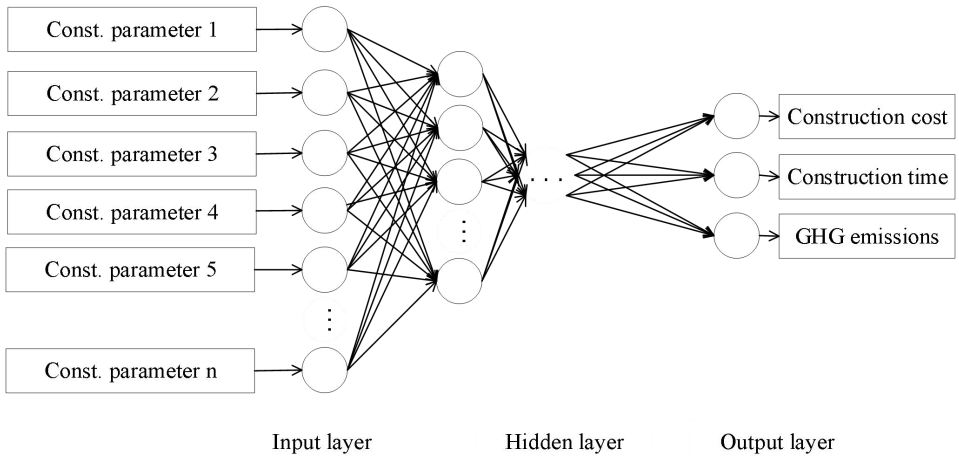

Figure 2: ANN based learning model for simulation-optimization integration.

2.3 Multi-Objective Optimization

Taking a balance of multi-objective performances is critical for construction engineering decisions. Particle swarm optimization (PSO) is a suitable tool for discrete variable decisions in construction. In this study, PSO is applied to solve construction multi-objective problems in proposed simulation-optimization framework.

The particle-updating mechanics of PSO was originally formulated by Eberhart and Kennedy(1995), and is revised by integrated with constructed learning model to evaluate multi-objective performances during optimization (see Figure 3). Each particle represents a possible construction plan, and these particles move along with a trajectory by Equation (3), update their position to their own best position and the global swarm of best position by Equation (4). The environment, cost, and time are defined as three elements in the particle position (*P*) and velocity (*V*) vector.

Where t=1,2,…,T denotes the iteration of PSO, whereas k=1,2,…,K denotes the particle of the swarm; $P^k(t)=\{C^k(t), T^k(t), E^k(t)\}$ denotes the position of the *k*-th particle in the *t*-th iteration for the three objective dimensions, whereas $V^k(t)=\{v_C{}^k(t), v_T{}^k(t), v_E{}^k(t)\}$ represents the velocity of position change; $pbestP^k(t\text{-}1)$ and $gbestP^k(t\text{-}1)$ describe the local and global best positions; inertia weight $w(t)$ describes the inertia of velocity influencing the *t*-th iteration by the *t*-1th iteration, which is defined as decreasing form as Wang et al. (2017) proposed; c_1, c_2 are learning factors (both set to 0.8), respectively; r_1 and r_2 are random numbers ranging from 0 to 1. The number of iterations (*T*) and population of particles (*K*) of PSO are set as 500 and 30, respectively. Two optimization stop criteria are set in this PSO-based optimization. The first is the maximum number of iteration, while the other is the certain number of iteration with improvement below threshold. If one of the criteria are reached, it implies the optimization has converged.

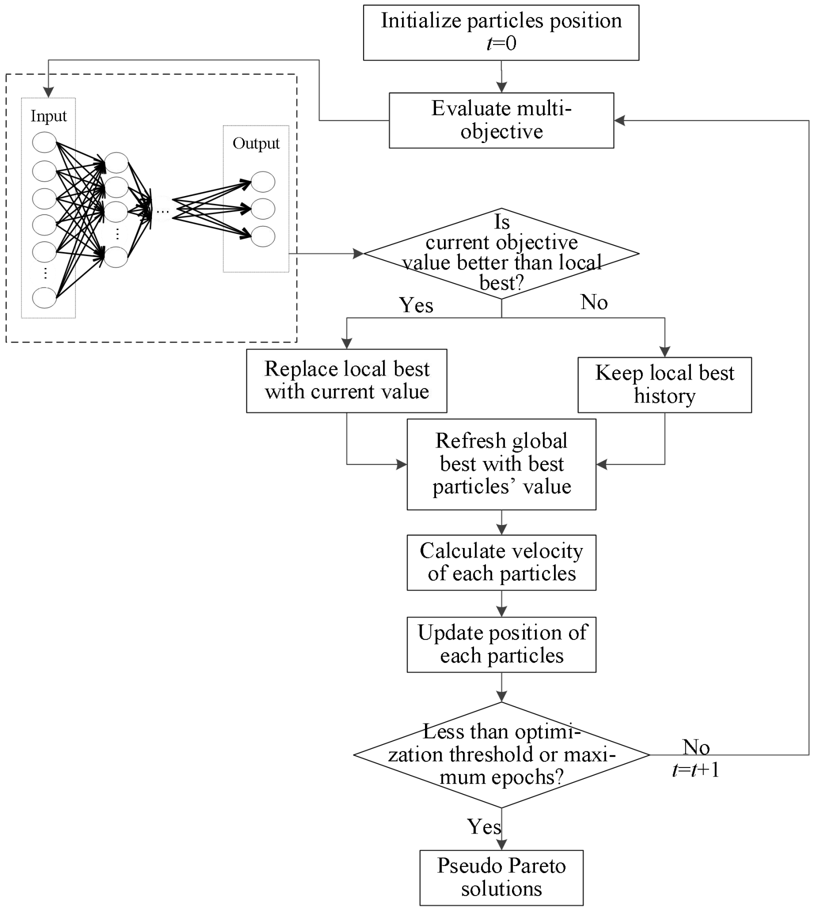

Figure 3: Machine learning based PSO optimization algorithms.

$$V^k(t)=w(t)\times V^k(t\text{-}1)+c_1\times r_1\times(pbestP^k(t\text{-}1)\text{-}P^k(t\text{-}1))+c_2\times r_2\times(gbestP^k(t\text{-}1)\text{-}P^k(t\text{-}1)). \quad (3)$$

$$P^k(t)=P^k(t\text{-}1)+V^k(t). \quad (4)$$

3 CASE STUDY

3.1 Discrete-Event Simulation

A hybrid concrete framework project including three residential buildings (as shown in Figure 4) is chosen as the study case to compare both the optimization performance and speed of analysis between MSO and SO. The project is located in Shenzhen, China. The total construction area of the project is 64050 m^2, and each of the buildings has 30 floors. In the project, most of the building components are using precast concrete (PC), such as external walls, internal walls, beams, slabs, balconies and stairs, while other concrete components are cast-in-situ (CS), including all the columns, part of beams and slabs. Based on the logistics and construction plans of the contractor, a Discrete Event Simulation (DES) model is built using software SIMIO.

Figure 5 shows the logic flow of a construction cycle in the DES model. A construction cycle contains the major construction activities of a typical floor in each of the three buildings and corresponding supply chains of PC components and CS materials. The interaction between the prefabrication activities of PC components and cast-in-situ activities of CS components, and the interaction between the construction and logistics are considered in the DES model. The PC components should arrive at the construction site before they could be hoisted and installed, and some PC components need to be installed before others. According to experiences, the arrived PC components have possibility of unacceptable quality that need to be returned (assumed as 10%). Sometimes the installation of PC components or supports is wrong and needs to be adjusted (assumed as 20%), which takes more time than the right installation. The concrete pouring will not start until all PC components, CS rebar and formwork of this floor have been installed. The construction of next floor will not be started before CS concrete in this floor has been cured for 12 hours.

Figure 4: The case project.

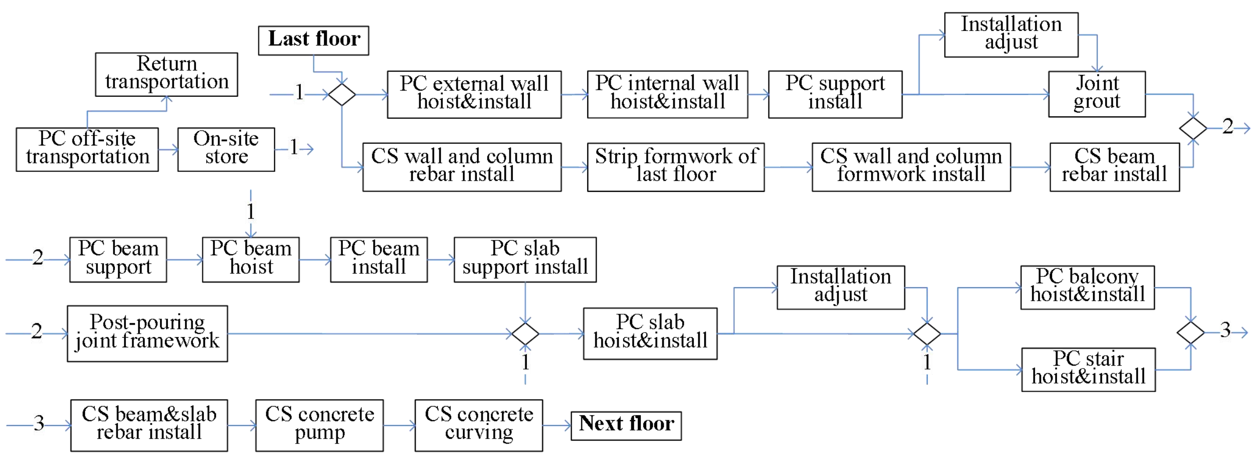

Figure 5: The logic flow of the DES model.

During building construction, the total construction time, the cost of labor, equipment and vehicle, and greenhouse gas emissions are three important indicators for contractors (Kamali and Hewage 2017; Mao et al. 2013). In order to calculate the above three indicators, the working time, productivity and electricity consumption of major construction equipment, the loading capacity and oil consumption of vehicles, and the productivity of labors need to be considered. Table 1 shows the Green House Gas (GHG) emission factors of electricity and diesel. According to the construction document of the studied project, alternative construction plans are found and listed in Table 2. The alternatives include number of PC trucks, PC installation workers, number and types of cranes, the supply chain mode of PC components, number and types of concrete pump, number of construction elevators and number of rebar processing workers. The supply chain has two modes: one is Just-In-Time (JIT) mode, under which the PC components are hoisted and installed right after they arrive construction site. Thus, there is no storage place for PC components on construction site. The other mode is the traditional transportation-storage-hoisting mode, under which the PC components are stored on the construction site before they are hoisted. The total optimization search space is 174,960 in this study, and it will be efficiently simulated and optimized by proposed MSO method. The minimum replication of simulation is determined as 26 according to trial-and-error method by Law and Kelton (2007).

Table 1: GHG emissions factors.

Impact Sources	**Unit**	**GHG emissions**	**Reference**
electricity	kg CO_2-e/kWh	0.714	National Development and Reform Commission 2011
diesel	kg CO_2-e/kg	3.153	Mao et al. 2013

Table 2: Alternative construction plans (total combination=174,960).

Construction tasks	**Alternative plans**	**Remarks**
PC wall transportation	8~12 trucks	30 t, 12.3 m×2.5 m
PC beam transportation	1~2 trucks	30 t, 12.3 m×2.5 m
PC slab transportation	3~5 trucks	30 t, 12.3 m×2.5 m
PC components installation	60, 70 or 80 PC workers	
PC components hoisting	2 cranes (STT293)	hoist time (minutes): PC=uniform(20.3, 22.3) CS=uniform(5.8, 9)
	2 cranes (XCP330HG7525-16)	PC=uniform(20.2, 22.3) CS=uniform(5.7, 9.1)
	1 crane (XGT8039-25)	PC=uniform(20.3, 21.8) CS=uniform(5.8, 8.5)
	1 crane (XGT500A8040-25)	PC=uniform(19.3, 21.4) CS=uniform(5.3, 8)
Concrete pouring	1~3 concrete pump (HBT6006A-5)	75 kW, 70 m^3/h
	1~3 concrete pump (HBT8016C-5)	132 kW, 85 m^3/h
	1~3 concrete pump (HBT6013C-5)	90 kW, 65 m^3/h
PC component supply chain mode	JIT	
	Transportation-storage-hoisting	
Rebar transportation	3~5 construction elevators (SC200/200)	66 kw, 2×2 t
Rebar processing	20, 30 or 40 rebar processing workers	

3.2 Machine Learning and Multi-Objective Optimization

After building of the DES model and setting the alternative construction plans into the DES model, the Machine Learning model needs to be built. In this study, the Artificial Neural Network (ANN) utilizing Bayesian Regularization is chosen as the Machine Learning method. The hidden layer size of the ANN model is 10. An amount of samples have been used to train the ANN model from the DES model. The sample number required for an accurate ANN model is got through trial and error, which will be discussed later in this paper. The samples include the input (the alternative construction plans) and output (construction time, cost and GHG emissions) of the DES model. The samples used to train ML model can be divided into two parts: training set and testing set. The training samples are used to train the ML. After training, the input of testing samples is given to ML model to generate the ML output, which will be compared with the output of testing samples. R square is the coefficient of correlation between learning model outputs and testing samples, which measures the accuracy of ML model to anticipate the outputs. The generated ML model need to be checked before it can be used as a substitute of the DES model. In this study, the ML model could be used in next step only when R square of the ML model has reached a required value, otherwise the model should be regenerated. The required value of R square will be discussed later in the Discussion & Conclusion section.

After the generated ML model satisfying the requirement, a multi-objective optimization should be used to find the pseudo Pareto solutions (the optimal construction plans of ML model) in MATLAB. In this study, the Particle Swarm Optimization (PSO) is used as the multi-objective optimization method. Table 3 shows parameters set in the PSO used in this study, which are proper PSO setting for construction optimization citing from previous study (Wang et al. 2017). Among the found optimal construction plans of ML model, there may still be some inferior plans which has inferior construction performance (construction time, cost and GHG emissions) in the DES model. Thus these found optimal plans should be input into the DES model to get their actual DES construction performance, based on which the PSO should be used again to select the real Pareto solutions out of the pseudo Pareto solutions.

Table 3: Parameters set in PSO.

Parameter	Value
Population number of each generation (N)	30
Number of iterations	500
Acceleration constant (c_1)	0.8
Acceleration constant (c_2)	0.8
Radom number (r_1 and r_2)	[0, 1]
Inertia weight (w)	[0.1, 1.2]

3.3 Results

The real Pareto solutions of MSO method with different number of samples are listed in Table 4, and the results of integrated DES model and PSO without Machine Learning are shown in the Table as the base scenarios. The results include the best indicators (time, cost and GHG emissions of a construction cycle) found in each scenario, R^2 and the difference between each scenario and the base scenario. As shown in Table 4, with the number of samples becomes larger, the best construction time, cost and GHG emissions found in Pareto solutions are getting close to those in the base scenario, and the R^2 gets larger.

The computing time by each method is also calculated in the MATLAB (shown in Table 5). As shown in Table 5, it costs 23391.66 seconds (6.50 hours) to get a result using SO, while it only takes 12070.31 seconds (3.35 hours) using MSO (125 samples, 26 simulation replications). In the MSO method, the time is mainly cost by ML training and finding pseudo Pareto solutions. The time cost by ML training increases when more samples are used to train the ML.

Table 4: Optimization results of best scenarios by SO and MSO.

Method (samples number)	**Time (day/floor)**	**Difference (%)**	**Cost (CNY/floor)**	**Difference (%)**	**GHG emissions (kg CO_2-e/floor)**	**Difference (%)**	**R^2**
SO	8983.84	/	8.81	/	187285.25	/	/
MSO (125)	8983.84	+0.00	8.84	+0.33	187285.25	+0.00	0.9986
MSO (100)	9009.59	+0.29	8.89	+0.83	188360.88	+0.57	0.9986
MSO (75)	9031.87	+0.53	8.84	+0.31	187285.25	+0.00	0.9962
MSO (50)	9004.61	+0.23	8.87	+0.63	190875.03	+1.92	0.9952
MSO (25)	10161.33	+13.11	8.84	+0.33	240947.31	+28.65	0.9486

Table 5: Computing time of SO and MSO method.

Method (samples number)	**ML training (s)**	**Finding pseudo Pareto solutions (s)**	**Re-simulation (s)**	**Finding real Pareto solutions (s)**	**Total**
SO	/	/	/	/	23391.66
MSO (25)	1662	81	5.87	0.29	1749.16
proportion (%)	95.01	4.63	0.34	0.02	100
MSO (50)	4248.5	78.77	79.11	0.4	4406.78
proportion (%)	96.4	1.79	1.80	0.01	100
MSO (75)	5943.2	84.61	316.19	0.4	6344.4
proportion (%)	93.68	1.33	4.98	0.01	100
MSO (100)	8750.27	79.19	137.37	2.05	8968.88
proportion	97.57	0.88	1.53	0.02	100

(%)					
MSO (125)	12070.31	78.77	1952.42	0.61	14102.11
proportion (%)	85.59	0.56	13.84	0.01	100

4 DISCUSSION AND CONCLUSION

The direct discrete-event simulation and optimization integration method is computationally expensive (Yousefi et al. 2018). To address this limitation, a machine learning based integration method was proposed and implemented in a construction case.

In the machine learning based method, major time is spent on training machine learning model (see Table 5). To achieve a reliable learning model for case project, e.g. optimization difference less than 1%, at least 75 samples are need to train and test embedded learning model. At this situation (see Table 4), the R square is approximate 0.9962, and the total simulation, training, and optimization time are 6344.40 (s). As the computing time of direct simulation-optimization integration (SO) is 23391.66 (s), (23391.66-6344.40)/23391.66=72.9% of the computing time is reduced by proposed method. Therefore, the proposed machine learning embedded integration method is demonstrated to significantly reduce (by 72.9%, compared to MSO with 75 samples) the computationally expensiveness of traditional simulation-optimization integration.

As for the performances optimization, the original SO method has better optimization performance compared with proposed method (see Table 4). Using proposed method in 75 training and testing samples, construction cost is 0.31% worse than SO method, and construction time is 0.53% worse than SO method, but GHG emissions has the same performance. Therefore, MSO method compromises optimization ability to reduce total computing time. This compromise is acceptable as new method only lose less than 1% of improvement, but it significantly reduced total computing time by 72.9%.

This machine learning based method should be more suitable for practical industry application, for which requires long time of simulation, or more optimization population and iterations, e.g. complex process interactions or great number of potential alternatives. For this situation, the traditional method, although, ensures a better optimization ability. Nevertheless, it costs too much computing time makes method itself industry impractical. The proposed machine learning embedded method, on the other hand, can be an alternative method for this application that compromises limited optimization performances, but it can provide support for decision-making in minutes rather than hours. Although the computing ability is increased with the development of cloud computing and super computer. They are not easily accessed for every engineering project and practitioner at present. Thus, the contribution of proposed method is to provide a more industry practical method for construction engineering to achieve simulation-optimization application.

The machine learning based construction simulation-optimization method MSO obtains reliable performances with significantly reduced computing loads. But the present study still has limitations that will be addressed in future study. The parameters setting of PSO-based optimization are from previous study. A sensitivity analysis could be valuable to reveal the proper optimization setting for proposed MSO method.

ACKNOWLEDGMENTS

This research was supported by Swedish Research Council for Environment, Agricultural Sciences and Spatial Planning (Formas); the Swedish Foundation for International Cooperation in Research and Higher Education (STINT).

REFERENCES

Eberhart R., and J. Kennedy. 1995. "A New Optimizer Using Particle Swarm Theory". In *Proceedings of the Sixth International Symposium on Micro Machine and Human Science*, October 4th–6th, Nagoya, Janpan, 39–43.

Fischer M., F. Aalami, C. Kuhne, and A. Ripberger. 1999. "Cost-loaded Production Model for Planning and Control". *Durability of Building Materials and Components* 8(4):2813–2824.

Hajjar D., and S. Abourizk. 2002. "Unified Modeling Methodology for Construction Simulation". *Journal of Construction Engineering and Management* 128(2):174–185.

Inyim P., and Y. Zhu. 2013. "A Framework for Integrated Analysis of Building Designs Using Life Cycle Assessment and Energy Simulation". In *Proceedings of the 2013 International Conference of Construction and Real Estate Management*, October 10th–11th, Karlsruhe, Germany, 316–327.

Kamali M., and K. Hewage. 2017. "Development of Performance Criteria for Sustainability Evaluation of Modular versus Conventional Construction Methods". *Journal of Cleaner Production* 142(2017):3592–3606.

Larsson J., W. Lu, J. Krantz, and T. Olofsson. 2015. "Discrete Event Simulation Analysis of Product and Process Platforms: A Bridge Construction Case Study". *Journal of Construction Engineering and Management* 142(4):04015097-1~12.

Law, A. M. and W. D. Kelton. (2007). *Simulation Modelling and Analysis*. Boston: McGraw Hill.

Li H., L. Zhang, D. Mah, and H. Yu. 2017. "An Integrated Simulation and Optimization Approach for Reducing CO2 Emissions from On-site Construction Process in Cold Regions". *Energy and Buildings* 138(2017):666–675.

Lu W., and T. Olofsson. 2014. "Building Information Modeling and Discrete Event Simulation: Towards an Integrated Framework". *Automation in Construction* 44(2014):73–78.

Mao C., Q. Shen, L. Shen, and L. Tang. 2013. "Comparative Study of Greenhouse Gas Emissions Between Off-site Prefabrication and Conventional Construction Methods: Two Case Studies of Residential Projects". *Energy and Buildings* 66(2013):165–176.

Monostori L., and Z. Viharos. 2001. "Hybrid, AI- and Simulation-supported Optimisation of Process Chains and Production Plants". *CIRP Annals* 50(1):353–356.

National Development and Reform Commission. 2011. *Provincial Greenhouse Gas Inventory Guidelines*. Beijing: NDRC.

Segerstedt A., and T. Olofsson. 2010. "Supply Chains in the Construction Industry". *Supply Chain Management: An International Journal* 15(5):347–353.

Shin Y., H. Cho, and K. Kang. 2011. "Simulation Model Incorporating Genetic Algorithms for Optimal Temporary Hoist Planning in High-rise Building Construction". *Automation in Construction* 20 (2011):550–558.

Wang Y., K. Feng, and W. Lu. 2017. "An Environmental Assessment and Optimization Method for Contractors". *Journal of Cleaner Production* 142:1877–1891.

Yadav B., S. Mathur, S. Ch, and B. Yadav. 2017. "A Simulation-optimization Approach for the Consideration of Well Clogging During In-situ Bioremediation System Cost Estimation". *Journal of Hydraulic Engineering* 23(3):1–11.

Yousefi M., M. Yousefi, R. Ferreira, J. Kim, and F. Fogliatto. 2018. "Chaotic Genetic Algorithm and Adaboost Ensemble Metamodeling Approach for Optimum Resource Planning in Emergency Departments". *Artificial Intelligence in Medicine* 84(2018):23–33.

Zhang H., and H. Li. 2004. "Simulation-based Optimization for Dynamic Resource Allocation". *Automation in Construction* 13(3):409–420.

AUTHOR BIOGRAPHIES

KAILUN FENG is a Ph.D. Candidate at the Department of Construction Management in Harbin Institute of Technology and in Luleå University of Technology. His research interests are construction simulation to assess and achieve project cost, time and environmental objectives. His email address is 15B910013@hit.edu.cn.

SHIWEI CHEN is a Ph.D. Candidate at the Department of Construction Management in Harbin Institute of Technology and in Luleå University of Technology. His research interests are integrating simulation and optimization to find optimal plans for prefabricated construction. His email address is chenshiwei97@163.com.

WEIZHUO LU is Associate Professor at the Construction Management and Building Technology Division of Industrialized and sustainable construction, Luleå University of Technology, Sweden. His research expertise in the area of sustainable construction, with a particular focus on how to use information technology to quantify the life cycle energy consumption and environmental impacts of building. His email address is weizhuo.lu@ltu.se .