

Embedding ensemble learning into simulation-based optimisation: a learning-based optimisation approach for construction planning

Embedding
ensemble
learning into
SO

259

Kailun Feng, Shiwei Chen, Weizhuo Lu, Shuo Wang, Bin Yang,
Chengshuang Sun and Yaowu Wang

(Information about the authors can be found at the end of this article.)

Received 5 February 2021
Revised 6 June 2021
Accepted 1 July 2021

Abstract

Purpose – Simulation-based optimisation (SO) is a popular optimisation approach for building and civil engineering construction planning. However, in the framework of SO, the simulation is continuously invoked during the optimisation trajectory, which increases the computational loads to levels unrealistic for timely construction decisions. Modification on the optimisation settings such as reducing searching ability is a popular method to address this challenge, but the quality measurement of the obtained optimal decisions, also termed as optimisation quality, is also reduced by this setting. Therefore, this study aims to develop an optimisation approach for construction planning that reduces the high computational loads of SO and provides reliable optimisation quality simultaneously.

Design/methodology/approach – This study proposes the optimisation approach by modifying the SO framework through establishing an embedded connection between simulation and optimisation technologies. This approach reduces the computational loads and ensures the optimisation quality associated with the conventional SO approach by accurately learning the knowledge from construction simulations using embedded ensemble learning algorithms, which automatically provides efficient and reliable fitness evaluations for optimisation iterations.

Findings – A large-scale project application shows that the proposed approach was able to reduce computational loads of SO by approximately 90%. Meanwhile, the proposed approach outperformed SO in terms of optimisation quality when the optimisation has limited searching ability.

Originality/value – The core contribution of this research is to provide an innovative method that improves efficiency and ensures effectiveness, simultaneously, of the well-known SO approach in construction applications. The proposed method is an alternative approach to SO that can run on standard computing platforms and support nearly real-time construction on-site decision-making.

Keywords Construction planning, Information and communication technology (ICT) applications, Optimisation, Simulation, Novel method

Paper type Research paper

© Kailun Feng, Shiwei Chen, Weizhuo Lu, Shuo Wang, Bin Yang, Chengshuang Sun and Yaowu Wang. Published by Emerald Publishing Limited. This article is published under the Creative Commons Attribution (CC BY 4.0) licence. Anyone may reproduce, distribute, translate and create derivative works of this article (for both commercial and non-commercial purposes), subject to full attribution to the original publication and authors. The full terms of this licence may be seen at <http://creativecommons.org/licences/by/4.0/legalcode>

The third author Shiwei Chen contributed largely to revise the manuscript. All authors would like to adjust Shiwei Chen to the second author. And Weizhuo Lu is the third and corresponding author of the article.

This project was funded by the China Postdoctoral Science Foundation (grant no. 2020M670918), the National Natural Science Foundation of China (grant no. 51878026), 2019 Education, Teaching and Research project of China Association of Construction Education (grant no. 2019087) and the Swedish Research Council for Environment, Agricultural Sciences and Spatial Planning (Formas).



1. Introduction

Plenty of decisions concerning schedule, equipment, resources, labour, supply chains and more need to be made in the construction planning of building and civil engineering projects. An optimisation method is useful for finding the best solution from various combinations of construction decisions (Wang *et al.*, 2017). The simulation-based optimisation (SO) framework, which integrates optimisation algorithms and simulation technologies, is a popular optimisation approach. In this approach, the simulation technology evaluates the performance, while optimisation such as meta-heuristic algorithm identifies the optimal solution (AbouRizk and Shi, 1994). It is consistently recognised that SO has excellent searching ability to identify the optimal solution from an extensive array of possible options (Gosavi, 2015). The provided optimal solutions are valuable references for construction planning (Feng *et al.*, 2018).

It is important to consider that construction operations are performed in a dynamic environment. This is because a construction project is a long-term endeavour in which the design, on-site conditions and available resources can change at any point (Collyer and Warren, 2009; Love *et al.*, 2002). These changes then shape the construction process by modifying the conditions for construction operations. Thus, the optimisation approach needs to continuously, and efficiently, adjust the optimal plan to the current situation (Al Hattab *et al.*, 2017). However, SO may not be an ideal optimisation approach in this situation because it is unable to provide a real-time response due to its high computational loads (Jun *et al.*, 2010; Nguyen *et al.*, 2014). This is mainly a result of how the optimisation framework works. Optimisation algorithms, especially the widely applied meta-heuristic algorithms, e.g. genetic algorithm (GA), particle swarm optimisation (PSO) and ant colony optimisation (ACO), normally leverage population advantage to reach the optimal solution (Magnier and Haghghat, 2010). This requires thousands, and sometimes more, fitness evaluations during optimisation. However, the simulation model will be invoked during every evaluation, which will require significant computing power. The slow response of SO approach is also partly explained by construction-specific features, with the most important being uncertainty (Segerstedt and Olofsson, 2010). Integrating the uncertainties into an optimisation evaluation usually requires repetitive simulation, which further increases computational loads of SO approach (Jun *et al.*, 2010; Lee *et al.*, 2015). Therefore, the SO framework normally takes days, and even possibly weeks, to provide a single optimisation result for construction decisions. This computational issue of SO, i.e. high computational loads causing the delayed response for every time need optimisation, makes it unsuitable tool to make frequent also timely decision support during construction operations.

Modification on the optimisation settings such as reducing searching ability, e.g., fewer particles or iterations are straightforward ways to address the computational load challenge. However, the optimisation quality is also offset by these settings due to less searching ability (Song and Gu, 2004). High-performance computing (HPC), including large-scale, cloud and parallel computing, is also able to address these issues through considerable computational capacity. But not all construction sites have the enabling environment or required network service for the necessary equipment (Apostu *et al.*, 2013). Therefore, it is necessary to develop an alternative optimisation approach for construction planning that can rapidly respond to changes but does not demand a high level of computational capacity.

From a statistical perspective, machine learning is the technology that is able to learn from data and make accurate predictions (Yan and Wang, 2014). The significant advantages are the real-time and accurate learning abilities, the features of which may help overcome above shortcomings of SO methods (Jun *et al.*, 2010). Furthermore, ensemble learning (EL) is an emerging machine learning mechanism that uses multiple learners for regression or classification (Breiman, 1996). The application of EL is beneficial because there is a low

likelihood of creating poor learning models for vastly different learning contexts (Zhang and Ma, 2012). Therefore, the EL is a promising algorithm to provide a generalised purpose learning method that can be applied to different construction projects that vary in scale, category and/or technology (Feng, 2019).

Therefore, this study proposed an optimisation approach for construction planning by embedding EL into the SO framework, i.e. EL embedded simulation optimisation (ESO), in which ensemble learning algorithms will develop an embedded information connection between simulation and optimisation technologies. The inherent real-time feedback ability and learning stability of ensemble learning ensures that this connection is both efficient and reliable. Specifically, the ensemble learning algorithm is trained, validated and tested by discrete-event simulation (DES), and will be integrated into the particle swarm optimisation algorithm (PSO) to be invoked during each fitness evaluation. This study investigates whether the developed ESO approach can reduce the intensive computational loads and ensure the optimisation quality for construction decision-makers with the both real-time and reliable optimisation method. Furthermore, the presented research investigates how the algorithm should be embedded and which settings are most appropriate in the context of project construction.

2. Literature review

2.1 Construction planning and optimisation

Plenty of planning decisions need to be made in the progress of building and civil construction. The optimisation method is a crucial tool for decision-makers to develop a preferred construction planning (Wang *et al.*, 2017). However, the construction is operated in a dynamic environment in which the design, on-site conditions and available resources can frequently change along with construction progress (Collyer and Warren, 2009; Love *et al.*, 2002). Thus, the decision-making in construction management should be adjusted frequently. In addition, the decisions in each construction stage are usually time-constrained due to the rushed schedule before executions (Yu *et al.*, 2005). The decision feedback in a fast time frame is greatly needed in this situation.

To provide timely support for the time-constrained construction decisions, it should both rely on capturing real-time project information and techniques to support efficient decision-making before activity execution (Tavakolan *et al.*, 2019). The emerging data collection technologies (e.g. Internet of things) make real-time information available. However, efficient decision-making support is still required. The usually applied SO is not an ideal approach because it has high computational loads that cannot provide timely feedbacks (Nguyen *et al.*, 2014), or unreliable optimisation quality at fewer computational loads settings (Song and Gu, 2004). This background motivates this research to develop an alternative approach with fewer computational loads for timely and reliable construction decision-making.

2.2 Simulation-based optimisation for construction

Discrete-event simulation (DES) is a simulation technology that conceptualises the status of target system changed after every occurrence of a discrete event (Cassandras and Lafortune, 2009). Construction processes are inherently complex due to their high degree of interaction, interrelatedness and uncertainty (Lu and Olofsson, 2014). This makes DES especially suitable for simulating construction operations and evaluating construction performance (Zhang and Li, 2004; Larsson *et al.*, 2016). Nevertheless, there are still only limited real-world applications of DES in construction (Lu and Olofsson, 2014). Previous research has recognised that the extensive computational loads of DES will impose limitations for its wide applications

(Budgaga *et al.*, 2016). In the construction context, the reason causing even higher computational loads to DES are the large array of scenarios in construction. The accumulated computational loads associated with the simulation of numerous scenarios makes DES impractical to analyse all alternatives of construction decisions (Feng *et al.*, 2018).

To avoid alternative enumeration in DES technology, SO is proposed (Cheng and Feng, 2003). The optimisation portion of the SO approach is used to identify optimal and near-optimal solutions from a huge array of possible construction scenarios, whereas the simulation is used to simulate and evaluate the construction performance of the identified solutions (Cheng and Feng, 2003). The SO method has been applied to many aspects of construction planning. AbouRizk and Shi (1994) performed one of the first SO investigations to determine optimal construction resource allocation. In their research, DES was integrated into optimisation iterations to identify the optimal resource allocation. Motivated by the optimisation ability of metaheuristics, Shin *et al.* (2011) proposed a DES model which incorporates a genetic algorithm (GA) to help contractors find the optimal hoist plans for high-rise building construction. They reported that the proposed method provided hoist plans that satisfied both the cost and time objectives.

2.3 Computational loads deficiency of simulation-based optimisation and potential benefits of machine learning

Even though SO is benefitted from optimisation with high searching ability, it is criticised not feasible for real-time decisions due to its still cumbersome computational loads (Nguyen *et al.*, 2014; Ninić and Meschke, 2015). Based on the SO framework, the optimisation algorithm will perform simulations during every fitness evaluation. The total computational load is affected by the optimisation population (N) and the number of iterations (I) required for convergence. This demonstrates why optimisation approaches that require simulation at every iteration ($N \cdot I$) are so time-consuming. Furthermore, a simulation will consume time (T) and require simulation replication (M) to provide a reliable result in the stochastic construction environments. Therefore, the total computational load of an SO method can be described as $N \cdot I \cdot M \cdot T$. As the processes associated with construction projects are complex and highly interrelated, the simulation time (T) is relatively high, which increases the total computational load ($N \cdot I \cdot M \cdot T$) to a level that is not feasible for real-time decision of construction projects (Nguyen *et al.*, 2014).

Reduction on the size of population of optimisation algorithms can balance the requirement of global optimisation and computational loads, which also indicates that population reduction will offset the optimisation quality (Song and Gu, 2004). To address the computational load challenge, several other valuable explorations have been conducted. Zhang (2008) proposed reducing the simulation replication (M) for non-optimal solutions. This was done by modifying a statistical method in SO that determines whether further replications are necessary. Parallel computing is another solution for reducing the computing time of SO. Yang *et al.* (2012) chose to run an optimisation in which simulations are computed in parallel. Thus, the computational loads of simulation replication (M) were reduced. On the contrary, Salimi *et al.* (2018) divided the population into multiple and set the optimisation to be paralleled in high-performance computing (HPC). This significantly decreased the population (N) parameter of computational loads. The parallel computing requires multiple cores or computers that act as platforms. As such, the computational loads stay the same, but the total computing time can be reduced as the computational loads are spread over different platforms. However, for some construction projects conducted under environments without enabling facilities or required network service, it usually lacks the resources required for parallel computing or other HPC. Therefore, it is valuable to explore the computational reduction alternatives to the SO methods that are suitable for the construction contexts.

The method presented in this study is motivated by the emergence of machine learning technology, which is characterised by both real-time feedback ability and high prediction accuracy (Colak and Qahwaji, 2009). A learning model could increase the efficiency of optimisation at every fitness evaluation because it avoids simulation time (T) and replication (M) by learning the construction performance from simulation dataset. It could therefore significantly reduce the overall computational loads by converting the SO's computational loads $N \cdot I \cdot M \cdot T$ into $N \cdot I \cdot t$ (the machine learning feedback time t is extremely smaller than $M \cdot T$). Moreover, this scale of machine learning does not require extensive computational capacity for model training and operation. Similar examples have already been applied, e.g. supply chain management (Jun *et al.*, 2010) and engineering design (Nakayama *et al.*, 2002), to reduce computational loads. However, the difference from previous research is that the interested parameters in construction contexts are both massive and changeable according to dynamic construction situations. Therefore, the proposed approach did not attempt to completely replace construction simulation, but rather embedded a machine learning algorithm within the SO framework to reduce overall computational loads also to provide reliable optimisation results.

2.4 Ensemble learning method

Machine learning comprises a set of algorithms that can learn from data to make real-time predictions (Yan and Wang, 2014). In all types of algorithms, the main goal of ensemble learning (EL) is to build a unified learning model that combines and takes advantage of different learning algorithms. To this end, EL generates multiple learners that are based on one (homogeneous approach) or different (heterogeneous approach) learning algorithms, and then provides a regression or classification by combining the outputs of every learner. Various studies have demonstrated that EL is more robust when applied to a problem in which a single learning algorithm shows significant performance variation (Zhang and Ma, 2012; João *et al.*, 2012). In this case, the ensemble learning approach will create a stable and generalised purpose learning model for ESO in the application of construction with different scale, category and/or technology.

Although EL clearly benefits data learning, its use is still at an early stage in applications related to AEC (architecture, engineering and construction). Only several researchers have attempted to solve AEC problems with approaches including EL. One example is the pilot study from Cao *et al.* (2018), in which EL was applied to predict highway project unit price bids. The results demonstrated that EL provides more stable and efficient unit price prediction than machine learning based on a single learning model. Wang *et al.* (2012) applied ensemble ANN to predict the cost and schedule success of construction projects. The dataset, which covered 92 projects, showed that ensemble ANN outperformed single ANN in terms of prediction accuracy.

EL approaches for ensemble generation and combination which employ multiple learners ensure more stable and generalised learning than approaches relying on the algorithm to develop a single learner (Webb and Zheng, 2004). The present study investigated whether the ability of real-time feedback, efficiency and generalisation of EL – when compared to the relatively computationally intensive SO framework and single learning method – can be harnessed in the context of a construction project.

3. Ensemble learning embedded simulation and optimisation

The ESO approach presented in the study comprises four modules: the modules of (1) simulation dataset generation, (2) EL model establishment, (3) learning-based optimisation and (4) Pareto solution refinement (see Figure 1). The module of simulation dataset generation

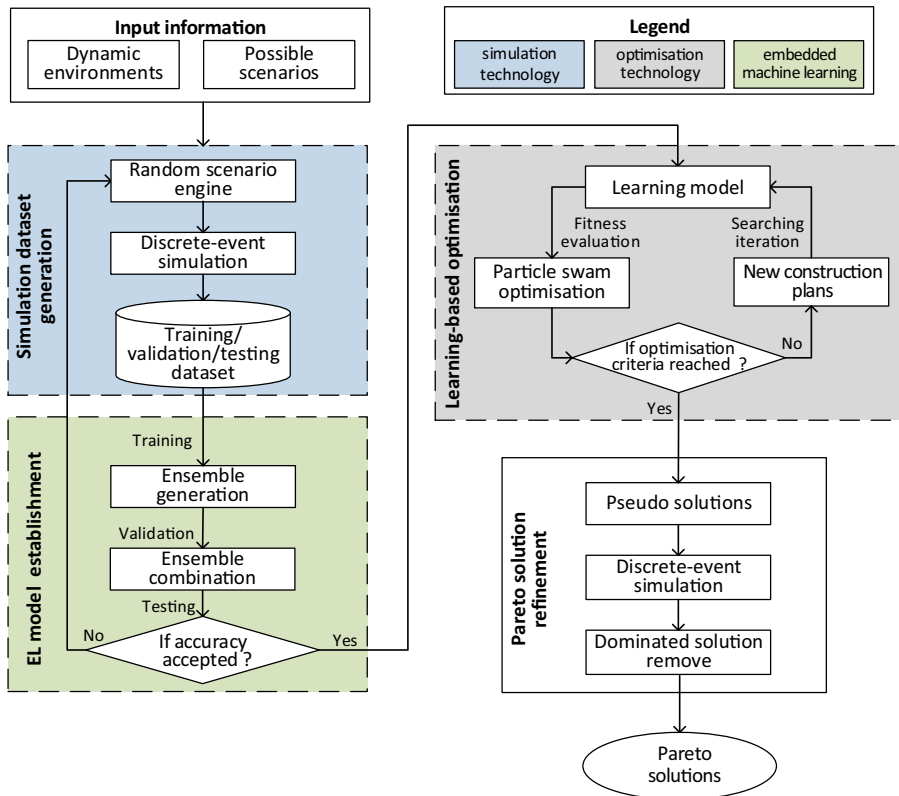


Figure 1. Overall framework of ensemble learning embedded simulation optimisation (ESO)

uses a random scenario engine based on Latin Hypercube Sampling (LHS) to generate construction scenarios in the scenario space. Then, the generated construction scenarios are then subjected to DES, which simulates construction performance. A dataset of construction plans, and the corresponding performance are thus built for embedded learning model training, validation and testing. After that, in the module of EL model establishment, multiple learners are trained using the training dataset, and these learners are then combined into a unified model using the validation dataset. The general performance of the EL model is tested using the testing dataset. A judgement procedure then increases the size of the construction dataset and re-builds the learning model until its accuracy is acceptable.

The established embedded learning model then integrates with PSO-based optimisation to perform a construction planning optimisation. The EL-based learning model enables a real-time and accurate fitness evaluation during each optimisation iteration. Hence, the presented ESO approach can theoretically reduce the extensive computational loads and ensure the optimisation quality compared with SO approach. The optimisation algorithm explores a wide range of construction scenarios until the defined stop criteria are reached and pseudo-solutions are obtained. A replication mechanism is set in the optimisation to enable the superior solutions without excessive increases in computational time. The Pareto solutions are finally obtained by removing the dominant solutions from the identified pseudo-solutions in the module of Pareto solution refinement. These solutions are timely and reliable optimisation support for construction decisions.

3.1 Simulation dataset generation

It is important to establish a specific and appropriate sampling method during the first stage of ESO framework, as it both should ensure the accuracy and efficiency of EL model development. A sampling engine majorly based on the LHS framework was employed to generate construction scenarios. The proposed random scenario engine meets the requirement of ESO, namely, it can generate a representative construction dataset with only limited size of samples. Thus, this random scenario engine will not cost too much time on sample generation and that may offset the advantage of efficiency. LHS samples various construction parameters, including construction technique, crew size and equipment, among others. Figure 2 demonstrates how LHS sampling can be applied to an example problem with two parameters and four samples. LHS sampling proceeds over two stages in this study. First, LHS divides construction parameter values into various portions characterised by equivalent probabilities, i.e. A1 to A4 and B1 to B4 in Figure 2. Then, the divided portions in the first stage are selected, and each portion in related dimension will be selected only once time. The parameter values in the selected portion are sampled by the original probability distributions, i.e. as $F_a(x)$ and $F_b(x)$. Therefore, LHS ensures that each scenario satisfies the original probability distribution and that the scenarios are not too centralised into a high probability space. LHS is a very classical method especially suitable for situations in which a limited number of samples are used to represent whole solution space (Eglaes and Audze, 1977).

To properly evaluate the performance of various construction scenarios, the proposed method simulated construction processes using DES. This decision was motivated by the fact that DES provides detailed process data that supports the evaluation of various construction aspects, such as cost, duration and greenhouse gas (GHG) emission estimations. The applied DES used construction plans as an input and provided construction performance as the output. These inputs and outputs are then used for EL model training, validation and testing. The developed DES model is validated by input-output validation proposed by Banks (1998). The DES model is considered valid if simulation results are statistically agreed with the actual project. The Wilcoxon signed-rank test is applied to test the agreement as many factors in construction are non-normally distributed.

An appropriate number of simulation replication is required for DES to ensure a satisfactory level of accuracy in the stochastic construction environments. To generate reliable samples for the EL model, the present study used the method by Lorscheid *et al.* (2012) to determine the number of simulation replications, which follows how the coefficient

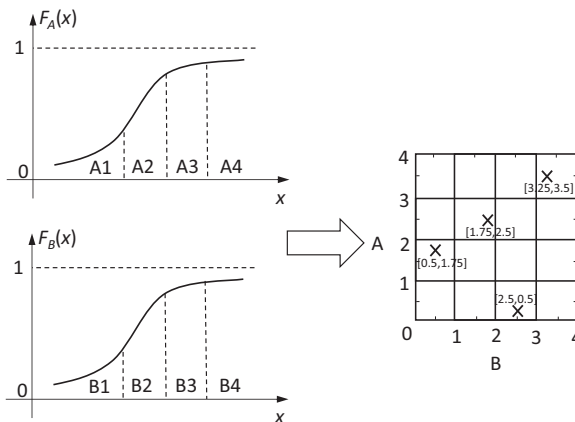


Figure 2. LHS two-stage sampling method illustration

of variation ($Cv = \text{standard deviation}/\text{mean value}$) changes as the number of replications (m) increase. Thus, the appropriate number of replications (n^*) that ensures a satisfactory level of accuracy can be identified by finding the point (minimum replications) at which the variation (Cv) across simulations becomes stable and below a specified threshold (E) (see Eq. (1)). Then, the simulation replication can provide the reliable evaluation, e.g. with average performance for construction dataset generation.

$$n^* = \arg \min \{n \mid |Cv^n - Cv^m| < E, \forall m > n\} \quad (1)$$

3.2 EL model establishment

3.2.1 Ensemble generation. An ensemble learning model (F) takes advantage of the diversity of multiple learners $\{f_1(\cdot), \dots, f_M(\cdot)\}$ to improve the generalisation and stability of prediction. The procedure for EL model development is depicted in Figure 3. The first step is the generation of a set of learners, i.e. ensemble generation. Bootstrap aggregating, first proposed by Breiman (1996), is the most widely used method for generating multiple learners in EL. In this method, the training data sets for each learner are formed by sampling from the total training data set with replacement. Another method for ensemble generation is boosting, which differs in the possibility of instance selection in training samples (Webb and Zheng, 2004). In boosting, instances with poor prediction have a higher possibility of being selected as training samples. According to Zhang and Ma (2012), bootstrap aggregating is more suitable for learning problems with limited training data. The present study applied bootstrap aggregating to generate EL learners mainly to save time on training data generation. It generates EL learners $\{f_1(\cdot), \dots, f_M(\cdot)\}$ by providing the training subsamples $\{S_1(\cdot), \dots, S_M(\cdot)\}$. The subsamples were generated by uniformly selecting samples from the total training data with replacement. According to previous studies (Erdal and Karahanoglu, 2016; Kim et al., 2003), the size of each subsample for individual learners training was set as

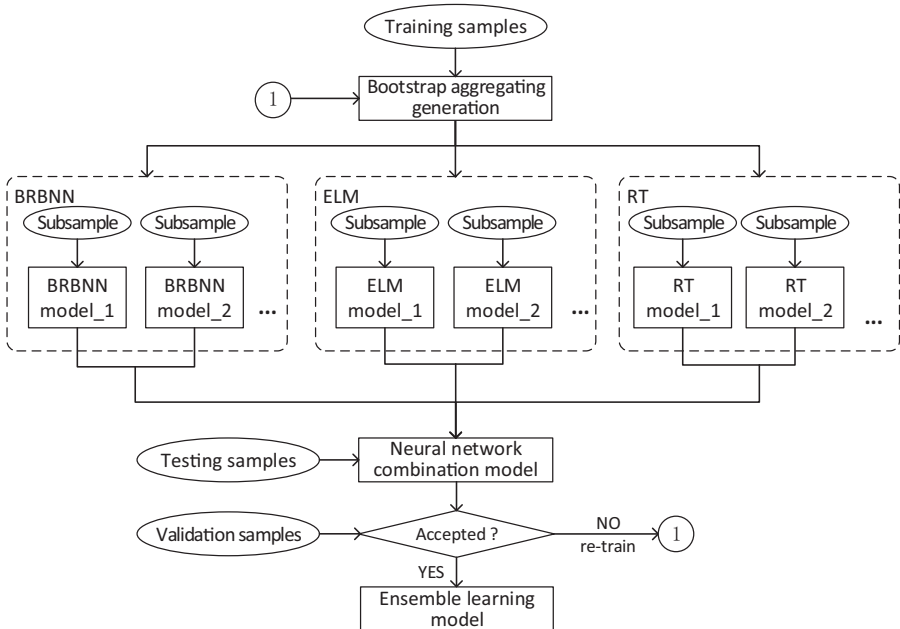


Figure 3. Ensemble learning development procedure

60% of the total training data. The appropriate number of learners depends on the problem context (Htike, 2016). In this study, the number of learners for each learning algorithm was preliminarily set at 20, and this number could be modified if learning accuracy was not acceptable.

The generalisation ability of the ensemble learning originates from multiple learners comprising it (Zhou, 2009). Hence, the selection of algorithms for multiple learners is critical for an EL model. Krogh and Vedelsby, 1995 formally demonstrated that the diversity of learners is a key factor for ensemble learning with a high degree of generalisability. The ESO developed in the present study uses a heterogeneous approach, i.e. a set of multiple algorithms, to integrate the EL model, as distinct algorithms inherently contribute to the total diversity of the final model (Webb and Zheng, 2004). Additionally, artificial neural networks and decision trees are two typical unstable algorithms that own high diversity, and they are suitable for EL model development (João *et al.*, 2012). These algorithms are also especially suitable for applied bootstrap aggregating as they show high sensitivity when trained with diverse training subsamples. Therefore, artificial neural networks and decision trees were selected as the machine learning algorithms for EL in this study.

When considering which specific algorithms from artificial neural networks and decisions trees to build EL, it is valuable to consider empirical evidence of their accuracy and feasibility in construction. Thus, they can both satisfy the diversity and the accuracy requirements for an ensemble learning (Krogh and Vedelsby, 1995). In the field of construction, the backpropagation neural network is a powerful and commonly applied example of an artificial neural network (Adeli, 2001). Rumelhart *et al.* (1985) claimed that the popularity of artificial neural networks partly owes to the development of backpropagation neural networks. The accurate learning ability of this method in the construction context has been validated by plenty of researchers over the years, with the earliest results tracing back to a 1997 construction scheduling study (Adeli and Karim, 1997). The application of Bayesian regularisation to backpropagation neural networks further extended the generalisability of machine learning (Foresee and Hagan, 1997). Extreme learning machine, unlike typical neural network algorithms, is an artificial neural network that chooses hidden nodes at random without the need for time-consuming iterative tuning (Huang *et al.*, 2006). In the field of construction, this approach has the advantage of acceptable learning accuracy with fast learning speed (Feng *et al.*, 2019). Hence, this algorithm could be especially suitable for building the large number of multiple learners required in EL. Another approach, regression trees, uses structural mapping of binary splits decision trees for regression problems (Breiman *et al.*, 1984). This approach has also previously been applied to construction projects, e.g., cost estimation of a project (Shin, 2015) and accident prediction (Liao, 2012), and demonstrated satisfactory accuracy. Based on the characteristics and advantages presented here, Bayesian regularisation backpropagation neural networks (BRBNN), extreme learning machine (ELM) and regression trees (RT) were chosen as the learning algorithms that would be integrated into the EL model presented in this study.

The basic idea underlying neural networks is represented in Eq. (2). BRBNN is based on backpropagation, which fits a neural network by calculating the gradient of the loss function and then iteratively tuning network parameters. The Bayesian regularisation of BRBNN uses Bayes' rules to infer the optimal regularisation parameters (Foresee and Hagan, 1997). On the other hand, ELM is a one-layer neural network that can choose hidden nodes randomly without the need for the time-consuming tuning of input weight w and bias b during each iteration. This algorithm only tunes the output weight β during model training. Thus, the main advantage of ELM is its fast learning ability.

$$\hat{y} = g \left(\sum_i^N W_i x_i + B_i \right) \quad (2)$$

where \hat{y} represents the predictive values of neural networks, x is the input of training samples, $g(\cdot)$ is the activation function of neural networks (Sigmoid function in this study), and W and B represent the network weight and bias vector, respectively.

RT initially establishes a tree that includes the full variable range found in the training samples (x_n, y_n) ($n = 1, \dots, N$). Then, it gradually splits into branches (t) in order to give a prediction function $f(x)$, such as the average of y ($y \in t$) for each sample belonging to t Eq. (3). Next, the splitting rule reduces prediction errors (S), as shown in Eq. (4), until the number of samples in every branch is below the minimum limit (Breiman *et al.*, 1984).

$$\hat{y} = f(x), x \in t \tag{3}$$

$$S = \frac{1}{N} \sum_{n=1}^N [y_n - \hat{y}_n]^2 \tag{4}$$

3.2.2 Ensemble combination. After learner generation, all of the learners should be combined as a unified learning model to form the final ensemble model that complement the different learning algorithms (i.e. ensemble combination). Applying the weight to each learner is the most commonly applied combination method (Breiman, 1996). Despite that, a machine learning based method has been proposed for ensemble combination (Yang and Browne, 2004). The present study used the machine learning based method – rather than the weighting method – for ensemble combination because this approach has demonstrated better results in problems in which there are complex, non-linear relationships between factors (Cao *et al.*, 2018), which may be suitable for construction performance modelling.

Previous EL research, as well as a prior pilot study in the construction field (Cao *et al.*, 2018), have validated neural networks as a good algorithm for ensemble combination. The logic underlying ensemble combination by neural networks is used and shown in Figure 4. The output of trained multiple learners based on training samples in ensemble generation serves as the input for combination learning model creation. The actual construction performances of the training samples are the output for the combination learning model. Hence, the neural network will connect the output of the trained multiple learners with the actual construction performance; in other words, this approach will combine multiple ensemble learners into a uniform model.

3.2.3 Ensemble learning evaluation. As the EL model is also designed to accurately connect simulation and optimisation, the proper indicators of its accuracy need to be established. The developed EL model will then be re-built until satisfactory prediction accuracy is achieved. Numerous indicators can be applied to evaluate the performance of a learning model, and a rational strategy is to use comprehensive indicators that can overcome the shortcomings of others. The coefficient of determination (R^2), normalised root mean square error (NRMSE) and mean average percentage error (MAPE) were utilised to evaluate EL model.

R^2 measures the goodness of fitness Eq. (5), with a higher R^2 value indicating that the EL model provides a sound representation of the simulation model. NRMSE is the normalised form of root-mean-square error Eq. (6), and measures the standard deviation between predicted and observed values, expressed as residuals. It can be especially useful for identifying significant deviation between the values predicted by the EL model and observations from the simulation. The normalised form of RMSE also ensures accurate comparison between models regardless of scale, which will be useful in evaluating how the model performs across different construction objectives. MAPE Eq. (7) describes the accuracy of prediction as a percentage. When compared to the other indicators, MAPE is easy to understand because the accuracy of prediction is represented by relative error. This indicator is widely applied in evaluations of regressions models developed by machine learning algorithms.

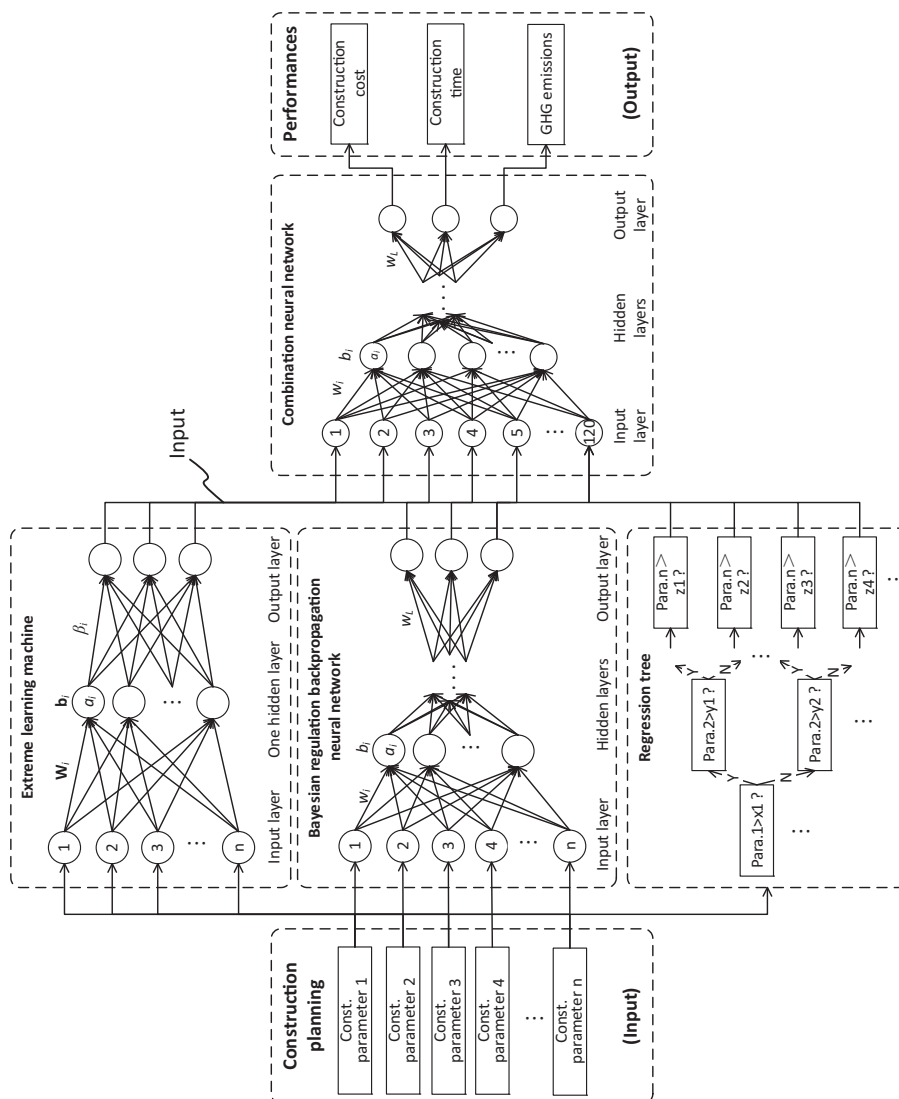


Figure 4. Ensemble learning-based construction performance modelling

$$R^2 = 1 - \frac{\sum_{t=1}^n (y_t - y)^2}{\sum_{t=1}^n (y_t - \bar{y})^2} \quad (5)$$

$$\text{NRMSE} = \frac{\sqrt{\frac{1}{n} \times \sum_{t=1}^n (y_t - y)^2}}{\max(y_t) - \min(y_t)} \times 100\% \quad (6)$$

$$\text{MAPE} = \frac{1}{n} \times \sum_{t=1}^n \left| \frac{y - y_t}{y_t} \right| \times 100\% \quad (7)$$

where n is the sample size, y_t is the t -th observed value, y is the value predicted by the learning model, and \bar{y} is the average value of y_t . The maximum and minimum values of an observed value are described by $\max(y_t)$ and $\min(y_t)$, respectively.

3.3 Learning-based optimisation

The developed EL model was then integrated into an optimisation algorithm to provide efficient evaluations during optimisation iterations. In this study, the developed EL model was integrated into a PSO algorithm to perform learning-based optimisation. PSO was originally formulated by [Kennedy and Eberhart \(1995\)](#), and is a suitable tool for multi-objective optimisation mainly due to fast convergence ability ([Kennedy et al., 2006](#)). Another advantage of using PSO for multi-objective optimisation is that this algorithm can readily achieve a non-weight multi-objective optimisation by defining multiple objectives as a vector in a particle position. The Pareto solutions that are non-dominated with each other on all objectives are obtained by PSO.

Based on the PSO mechanism, EL model will be invoked during each fitness evaluation. Then, the particle swarm will refresh or keep the global and local best information based on fitness results. The particle velocity will be updated, and particles will search new construction scenarios. In this way, the particle swarm will move to positions (scenarios) that show better performance by using information from previous individual positions, as well as the total swarm's positions.

The ideal size of the PSO population depends on the problem that is being optimised ([Bratton and Kennedy, 2007](#)). [Rohler and Chen \(2011\)](#) verified that a reasonable PSO population ranges from 10 to 40. In this study, a sensitivity analysis was conducted to identify the appropriate PSO population and performance tendency in the ESO framework. According to a previous study ([Feng et al., 2018](#)), the 500 PSO iterations applied in this study is an acceptable amount for construction planning optimisation when considering aspects of quality and time. As the ESO approach provides efficient optimisation, a replication mechanism can help the algorithm yield superior results with only limited computational loads. For this reason, a replication mechanism was set for learning-based optimisation. The optimisation quality is the measurement of searched Pareto solutions when compared with other set of Pareto solutions. The acceleration constant and inertia weight values were initially set based on previous construction applications ([Wang et al., 2017](#)), and will be modified if necessary (see [Table 1](#)).

3.4 Pareto solution refinement

The Pareto solutions will be obtained by determining the exact performance of pseudo solutions and removing solutions that are dominated by others. The inherent knowledge of

the relationship between construction planning and project performance contained in the EL model ensures that the searching trajectory of ESO converges to actual optimal solutions. However, these searched solutions are still *pseudo*-solutions because the fitness evaluation is based on a learning model. In order to determine the exact performance of pseudo solutions, the inputs of these solutions, i.e. the considered construction parameters, are provided to the DES model to simulate. Then, after an evaluation of the exact performances, the dominated solutions are removed and the remaining solutions are Pareto solutions.

4. Optimisation evaluation, prototyping and application

4.1 Optimisation evaluation

Evaluation indicators will be used to evaluate and compare the proposed ESO approach with the classical SO method. The ultimate objective of both of these methods is multi-objective optimisation. Thus, the optimisation quality should be evaluated based on the identified Pareto construction solutions. The hypervolume comparison method, proposed by Zitzler *et al.* (2003) and described by Eq. (8), can quantitatively compare the performance of different Pareto fronts. In this study, the method was used to evaluate the construction solutions obtained through different optimisation methods. The hypervolumes for the solutions of two optimisation approaches applied to a two-objective problem are illustrated as Figure 5. The hypervolume indicator can detect the situation that one Pareto front (*A*) is not worse than another front (*B*) when every solution in *B* is weakly dominated by at least one solution in *A*. This means that the comparison is capable of evaluating the Pareto front regardless of the number of solutions.

Parameter	Value
Population of each generation (<i>N</i>)	10, 20, 30, 40
Maximum number of iterations	500
Acceleration constant (<i>c</i> ₁ , <i>c</i> ₂)	0.8
Random number (<i>r</i> ₁ and <i>r</i> ₂)	Uniform [0, 1]
Inertia weight (<i>w</i>)	dynamic value $w_{\max} - (w_{\max} - w_{\min}) \times t/T$ $w_{\min} = 0.1$ and $w_{\max} = 1.2$

Table 1.
Parameters setting in
PSO-based
optimisation

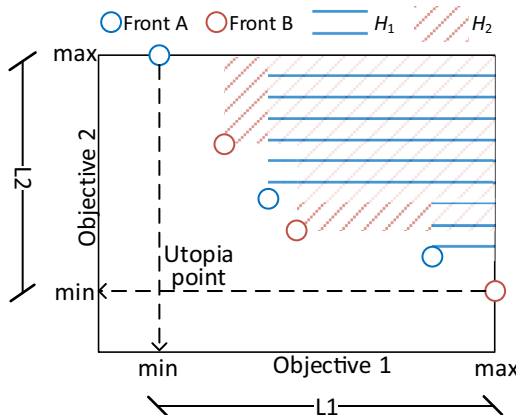


Figure 5.
An illustration of the
hypervolume
comparison of Pareto
fronts provided by two
optimisation models
that were applied to a
two-objective problem

$$NH_i = H_i / \prod_{\substack{\text{objective}=j \\ \text{objective}=1}}^{\text{objective}=J} (\max_j - \min_j) \quad (8)$$

where i indicates the evaluated Pareto front, j is the specific evaluated objective, H_i is the obtained hypervolume by i -th Pareto front and NH_i is the normalised hypervolume indicator that represents the optimisation quality of i -th Pareto front to the utopia point, while \max_j and \min_j represent the maximum and minimum values, respectively, of all evaluated Pareto fronts on j -th objective.

4.2 Method prototyping

A prototype was developed based on the proposed ESO approach to validate model efficiency and effectivity. The discrete-event simulation in ESO was developed within the SimioTM platform (version 10; Simio LLC, Sewickley, PA), while the EL model and multi-objective PSO were programmed in Matlab^R (version R2017a; Mathworks, Natick, MA). Data exchange between simulation, optimisation and learning models was facilitated by programming an API framework. Most importantly, all of the computing was performed on an office-level laptop with an Intel(R) CPU i5-8300H, 2.30 GHz with 4 cores. Prototype will execute automatically of ESO approach, and progressed from data generation, model training and model testing to planning optimisation (Figure 6). The detail pseudocode for the ESO framework can be found in Appendix 1.

4.3 Application of the developed ESO

The presented ESO approach was developed as a computational reduction approach alternative to the conventional SO method. The decision to embed an EL model within the SO framework was motivated by the need for an efficient connection between the simulation and optimisation modules to reduce the computational loads associated with the conventional SO framework (see Figure 7).

In order to apply the developed ESO approach to a construction project, the decision-maker should start by developing the simulation model and programming the optimisation algorithms, as is the case in the conventional SO framework. Next, an EL algorithm will be programmed and embedded between the simulation and optimisation modules of the SO framework, as shown in Figure 1. The decision-maker can select the preferred construction alternatives for optimisation, and they should be defined and manipulated in the simulation model just like SO framework along with the project progress. After these steps, the ESO is ready to run the construction optimisation.

Ideally, the ESO will provide a final Pareto front of solutions that compares to what would be identified by SO, yet at noticeably lower computational loads. This means that ESO could be run on the standard computing platforms already available at construction sites, i.e. personal desktops and laptops. Therefore, the benefits of ESO are most obvious when real-time optimisation support are required but HPC is not available.

5. Case application

5.1 Case information

The presented optimisation method and prototype were applied to a real-life large-scale construction project located in Shenzhen, China, to test its performance to the construction optimisation. The selected construction project includes three residential buildings (total area 64,050 m², 30 floors) with a reinforced concrete framework. This project was chosen because, on one hand, the project duration, estimated at more than two years. Thus, the project

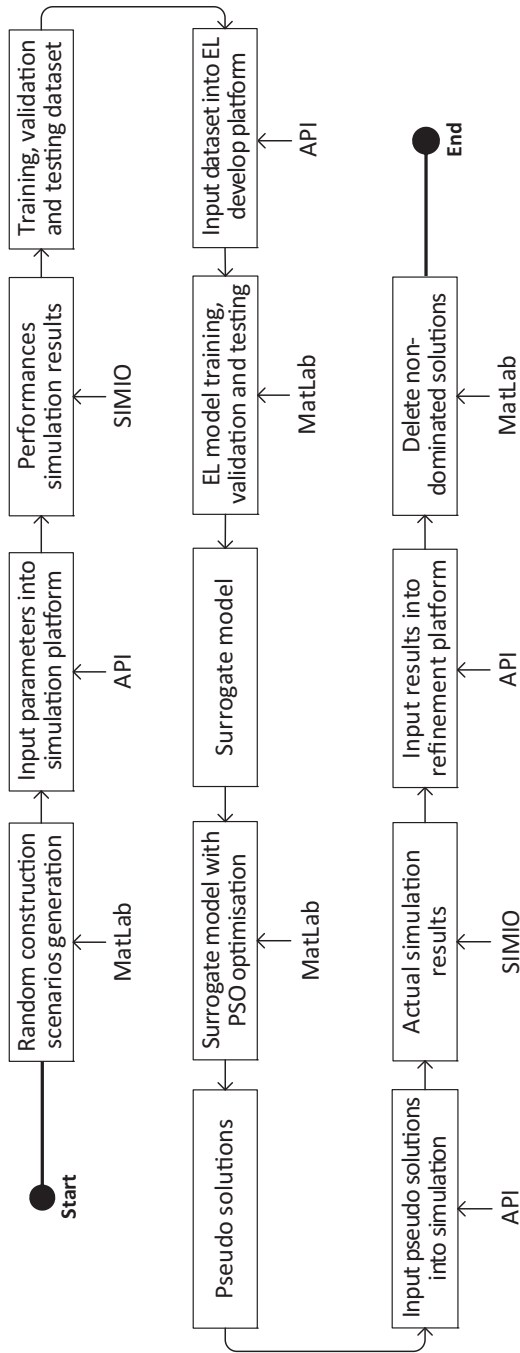


Figure 6.
The overall framework
of the ESO prototype

environment may vary multiple times during the project duration, which will influence the optimality of the original plan determined at the planning stage. Moreover, the project includes complex processes because the construction will use both precast (PC) and cast-in-situ (CS) construction. The external walls, internal walls, beams, slabs, balconies and stairs are PC components, while other components, including all of the columns and parts of beams and slabs, will be constructed using CS method. The PC and CS activities cause interaction with each other during construction; therefore, it is a challenge to identify the optimal construction scenario. The above features make this case suitable to apply and compare the proposed ESO with conventional SO method.

The processes involved in standard floor construction of the case building, depicted in Figure 8, are surveyed according to construction planning documents and the on-site manager. These processes include construction activities and involve the corresponding supply chains of PC components and CS materials. As such, efficient floor construction must consider the interaction between PC and CS construction, as well as the interactions between construction and material supply logistics. Based on Figure 8, the PC components should arrive at the construction site before they can be hoisted and installed. However, a certain share of the PC components will be returned if they do not pass the quality check. Furthermore, certain PC components need to be installed before other PC components. According to the on-site manager's expectation, the installation of PC components and temporary supports needs to be adjusted based on project-specific circumstances, which makes part of PC installation slower than predefined performance. To make the situation even more complex, the concrete pump cannot be started until all PC components, CS rebar and formwork of the floor have been installed, and the construction of the next floor will not start before the CS concrete on this floor has been cured for 12 hours.

Dynamic factors of the construction, such as available equipment (R_1) and crew (R_2) could change as construction progresses. Additional dynamic factors include the share of unacceptable PC materials and installation productivity, while unexpected situations can arise during the construction process. Thus, it can be expected that several optimisation runs will be needed to adjust the previous optimal solution to the current project environment. The frequency of adjustments will increase if the project occurs in a highly dynamic environment, which is usually the case for construction projects. In the investigated case, the initial plan for standard floor construction was based on managers' experiences from previous project environments. As such, the managers expected to have 100 and 60 workers available for PC component installation and CS rebar processing, respectively. However, at the start of standard floor construction, the managers only had 80 and 40 workers available for PC and CS processes, respectively. This was the result of one of the labour firms cancelling the contract with the project contractor due to cooperation failures. Therefore, the original optimal plans needed to be immediately adjusted to the new project environments before standard floor construction could be started.

The alternative plans for standard floor construction after the construction document review and discussion with the on-site managers are listed in Table 2. The new number of on-site workers can also influence the supply chains and equipment. Therefore, the alternatives

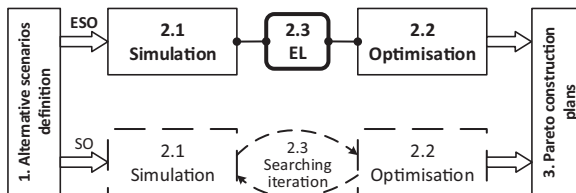


Figure 7.
A comparison of the
ESO and SO
application procedures

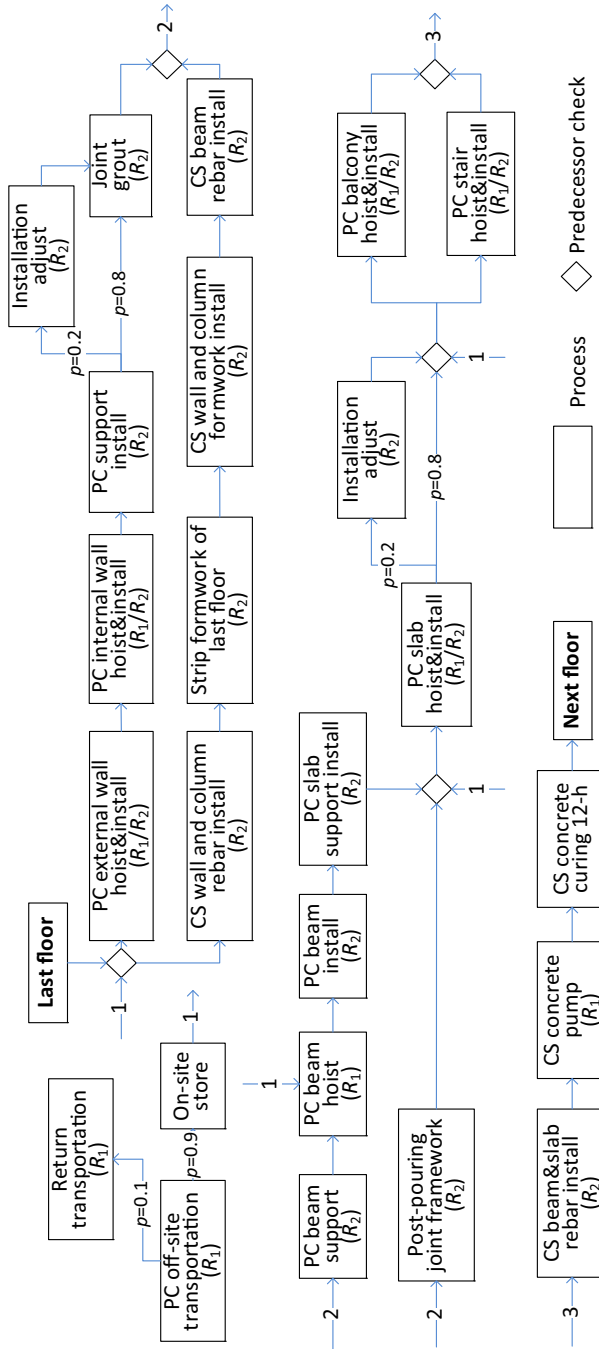


Figure 8. The logic flow of the standard floor construction in terms of equipment (R_1) and labour resource (R_2)

Construction task	Alternative scenario	Remarks
PC components installation	Up to 80 workers	
CS rebar processing	Up to 40 workers	
PC wall transportation	8~12 trucks	30 t, 12.3 m × 2.5 m, 37 litre diesel/100 km, transport Time ~ Uniform(100–120) min
PC beam transportation	1~2 trucks	30 t, 12.3 m × 2.5 m, 37 litre diesel/100 km, Time ~ Uniform(100–120) min
PC slab transportation	3~5 trucks	30 t, 12.3 m × 2.5 m, 37 litre diesel/100 km, Time ~ Uniform(100–120) min
CS concrete pouring	Concrete pump (HBT6006A-5) 1~3 pumps	75 kW, 70 m ³ /h
	Concrete pump (HBT8016C-5) 1~3 pumps	132 kW, 85 m ³ /h
	Concrete pump (HBT6013C-5) 1~3 pumps	90 kW, 65 m ³ /h
CS rebar transportation	3~5 Construction lifts (SC200/200)	66 kW, 2 × 2 t
PC components hoisting	1 crane (XGT8039-25)	Hoist time (min) PC = Uniform(20.3, 21.8) CS = Uniform(5.8, 8.5) hoist motors power(kW): PC = 58.1, CS = 37.6
	2 cranes (XCP330HG7525-16)	Hoist time (min) PC = Uniform(20.2, 22.3) CS = Uniform(5.7, 9.1) hoist motors power (kW): PC = 49.8, CS = 32.2
	2 cranes (STT293)	Hoist time (min) PC = Uniform(20.3, 22.3) CS = Uniform(5.8, 9) hoist motors power (kW): PC = 55.6, CS = 36.0
	1 crane (XGT500A8040-25)	Hoist time (min): PC = Uniform(19.3, 21.4) CS = Uniform(5.3, 8) hoist motors power (kW): PC = 74.7, CS = 48.4
PC component supply chain mode	Just-in-time (JIT) transportation-storage-hoisting	Supply chain without on-site storage Store one floor of PC component on-site
The alternative construction scenarios	Total possible scenarios	20,736,000

Table 2.
The alternative construction scenarios

also include appropriate numbers of PC transport trucks, concrete pumps, construction lifts and cranes, as well as the PC supply chain mode. All these alternatives can be adjusted before the construction of standard floor begins. The supply chain has two modes, the first being just-in-time (JIT) mode, in which the PC components are hoisted and installed immediately after arrival on-site. In this mode the PC components cannot be stored at the construction site. The other mode is the traditional on-site storage mode, in which the PC components are stored at the construction site before they are hoisted. Theoretically, JIT mode provides savings in terms of on-site re-transportation time, cost and emissions, while storage mode reduces the risk of off-site transportation delays.

The productivity of equipment, along with rental cost and energy productivity, was surveyed from equipment properties, a local Construction Engineering Quota database (SCPCM, 2016b, 2016a), and contractors' previous records. Whenever the project environment changes, contractors need an efficient tool that will provide optimal plans for

their on-site decisions. The presented ESO approach and conventional SO were applied and compared in this case project.

Finding a balance between multi-objectives is a critical part of construction decisions (Feng *et al.*, 2018). Project duration and cost are two most commonly considered objectives in construction optimisation (Eshtehardian *et al.*, 2009; Zhang and Ng, 2012). However, the negative effects of GHG emissions, i.e. global warming, have caused construction managers to include environmental performance in their evaluations of construction operations (Kamali and Hewage, 2017; Mao *et al.*, 2013b). This case project will be the tallest high-rise building with an integrated shear wall structure in southern China; therefore, including environmental perspectives in the optimisation can serve as a benchmark for future construction projects. In this case, the construction duration, the cost for labour and equipment, as well as construction-associated GHG emissions were the multi-objective indicators. The productivity and electrical/fuel consumption of the construction equipment and vehicles, along with the productivity of labourers (see Table 2), were included in the developed DES model. Table 3 shows the GHG emissions and costs of the electricity and diesel consumed during the construction operations.

5.2 Setup of ESO approach

The logic flow of the standard floor construction, including all tasks, associated resources and efficiencies, are developed into a construction DES model (details see Table 4). The input-output validation method is applied to validate the developed DES model. The input of simulation is the original planning in construction documents. The output of the simulation is the simulated results of the construction duration, the cost and GHG calculation, which are compared with actual performances planned in documents and quantity calculation (i.e. theoretical values). The DES model is validated, and the results are shown in Appendix 2.

And the appropriate simulation replication that ensures a satisfactory level of simulation accuracy under uncertainty is determined by coefficient of variation (Cv) method of Lorscheid

Source	GHG or cost factor	Source(s)	GHG or cost factor
electricity (kg CO ₂ -e/kWh)	0.714 (NDRC, 2011)	labour for joint grout (CNY/8 h, the same hereafter)	159.62 (SCPCM, 2016b, 2016a)
diesel (kg CO ₂ -e/kg)	3.153 (Mao <i>et al.</i> , 2013a)	labour for all CS transport	10005.194
cranes STT293 (CNY/day, the same hereafter)	979.07 (SCPCM, 2016b, 2016a)	labour for PC stair hoist and install	3583.56
cranes XCP330HG7525-16	845.88	labour for PC beam hoist and install	2875.69
cranes XGT8039-25-16	1445.22	labour for PC slab hoist and install	2342.88
cranes XGT500A8040-25	1445.22	labour for CS concrete curving	475.077
concrete pump HBT6006A-5	1883.15	labour for PC internal wall hoist and install	37.287
concrete pump HBT8016C-5	2,248.00	labour for PC balcony hoist and install	9.508
concrete pump HBT6013C-5	1761.53	labour for PC external wall hoist and install	118.821
construction lift SC200/200	431.13	labour for CS wall and column rebar transport	4783.303
–	–	labour for CS beam and slab rebar transport	5986.582

Table 3.
GHG emissions and
cost factors

Task name	Resource: Equipment (R_1)/worker (R_2)		Unit	Efficiency
PC external wall hoist	Crane (STT293)	PC worker	min/ pcs	Uniform(20.3, 22.3)†
PC internal wall hoist	Crane (STT293)	PC worker	min/ pcs	Uniform(20.3, 22.3)†
PC wall support install	Crane (STT293)		min/ pcs	Uniform(5.8, 9)†
Installation adjust	PC worker		min/ pcs	$P\{20\} = 20\%$ $P\{0\} = 80\%$ †
Joint grout	Rebar install worker		min/ pcs	15.024*
PC beam hoist	Crane (STT293)	PC worker	min/ pcs	Uniform(20.3, 22.3)†
PC slab hoist	Crane (STT293)	PC worker	min/ pcs	Uniform(20.3, 22.3)†
PC stair hoist	Crane (STT293)	PC worker	min/ pcs	Uniform(20.3, 22.3)†
PC balcony hoist	Crane (STT293)	PC worker	min/ pcs	Uniform(20.3, 22.3)†
PC beam support	Crane (STT293)		min/ pcs	Uniform(5.8, 9)†
PC slab support install	Crane (STT293)		min/ pcs	Uniform(5.8, 9)†
PC wall transportation	Wall truck		min/ batch	Uniform(100, 120)†
PC beam transportation	Beam truck		min/ batch	Uniform(100, 120)†
PC slab transportation	Slab truck		min/ batch	Uniform(100, 120)†
PC stairs and balcony transportation	Stairs and balcony truck		min/ batch	Uniform(100, 120)†
PC wall unload	Crane (STT293)	Wall truck	min/ pcs	Uniform(7.3, 9.3)†
PC beam unload	Crane (STT293)	Beam truck	min/ pcs	Uniform(7.3, 9.3)†
PC slab unload	Crane (STT293)	Slab truck	min/ pcs	Uniform(7.3, 9.3)†
PC stairs and balcony unload	Crane (STT293)	Stairs and balcony truck	min/ pcs	Uniform(7.3, 9.3)†
CS concrete pump	Pump		m ³ /h	70
CS beam and slab rebar install	Rebar install worker		min/m ²	10.04*
CS wall and column rebar install	Rebar install worker		min/m ²	11.63*
Post pouring joint formwork install	Crane (STT293)		min/ pcs	CS = Uniform(5.8, 9)†
CS wall and column and beam formwork install	Formwork worker		min/m ²	13.85*
CS wall and column rebar processing	Rebar processing worker		min/m ²	60.26*
CS wall and column rebar transportation	Rebar processing worker	Lift	min/ batch	(Uniform(1, 2) × ht./ 36 + 6)/4*
CS beam and slab rebar transportation	Rebar processing worker	Lift	min/ batch	(Uniform(1, 2) × ht./ 36 + 6)/4*

Table 4.
The tasks, associated
resources and
efficiencies in
DES model

Note(s): * the value is calculated from local Construction Engineering Quota database (SCPCM, 2016a, 2016b),
† the value is from on-site pre-experiments

et al. (2012), as Eq. (1). Figure 9 demonstrates that for the investigated case, the *Cv* of construction time, cost and GHG stabilises (under 0.1% difference) when 26 or more replications are performed. Thus, the minimum number of DES replications for the case study was set at 26. The parameters in the proposed ESO include structure of multiple/combination learners, number of samples and number of particles in PSO. And all of them will be analysed in following sections. The values for parameters in ESO are summarised in Table 5.

To test that the EL-based learning model can accurately extract construction information from DES, firstly, 800 samples were used to train, validate and test the EL model. The regression plots of actual versus predicted values for GHG emissions, time and cost of the testing samples are shown in Figure 10. The R^2 and coefficient a ($y = ax + b$) for all three objectives are close to 1. According to the standard presented in Gilliland’s research (Gilliland, 2010), the MAPE results for all three construction objectives could be regarded as “good” because all values fell below 20%. And as for NRMSE, no widely agreed standard exists for the field of construction. However, based on Hall’s research (i.e. $NRMSE < 5\%$), the time and cost performance are both “reasonable”, while GHG performance is slightly below the standard (Hall *et al.*, 2014). These results demonstrate that the developed EL model can accurately predict construction performances and, as such, can serve as an accurate connection between the simulation and optimisation modules in developed ESO approach.

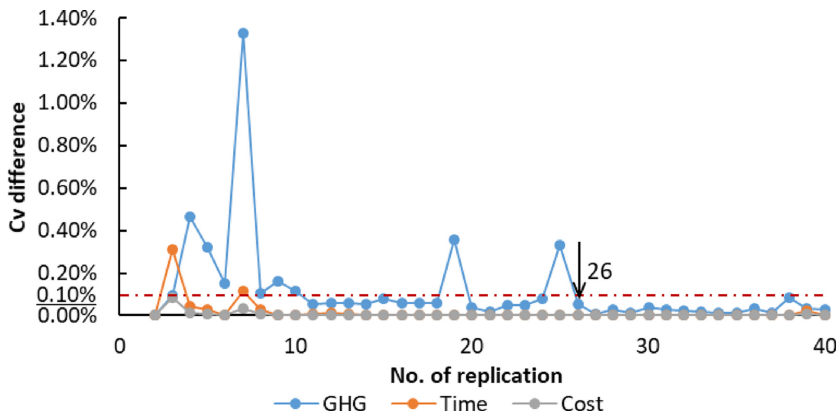


Figure 9. Coefficient of variation (presented as % difference) for the number of simulation replications

Parameter	Value
Total number of learner (<i>M</i>)	20:20:20
Structure of multiple learners	BRBNN: 10-3-3; ELM: 10-10-3 RT: min. leaf and parent are 1 and 10
Structure of combination learners	Compare among 120-3-3 (default), 120-10-3 and 120-10-10-3
Number of samples	Adapted increase from 50, 100, 200, 400 and 800
Rate of training sample	70%
Rate of testing sample	15%
Rate of validation sample	15%
Size of subsample for training in total training set	60%
Number of particles in PSO	Compare among 10, 20, 30 and 40

Table 5. Values for the parameters in the presented ensemble learning model

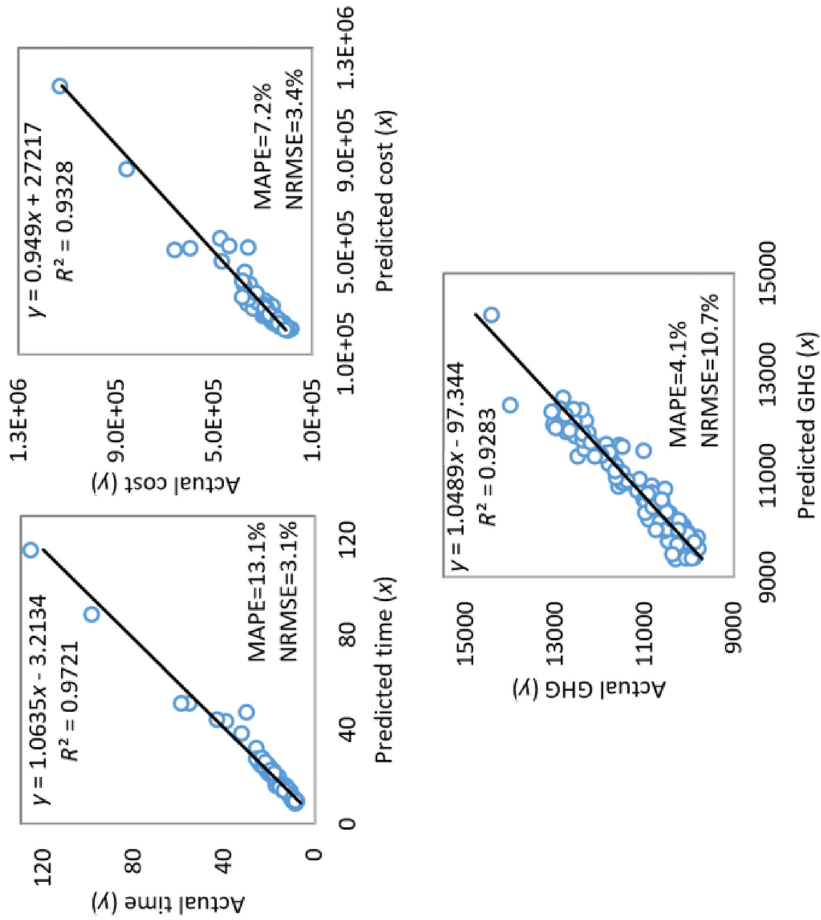


Figure 10. Regression plots of actual vs predicted values by ESO in (a) time, (b) cost, (c) GHG per floor construction

6. Results and discussion

6.1 ESO parameter and SO comparative analysis

The applicability of the presented ESO framework to construction projects was tested by analysing how changes in certain model parameters affect optimisation performance. The normalised hypervolume indicator NH was used to represent the normalised optimisation quality, with variation (Var.) and average improvement (AI) defined by Eqs. (9) and (10). The results illustrated in Figure 11 imply that two features of ESO affect optimisation performance. The first is the sample size used for model training, validation and testing; this factor influences optimisation performance because it determines how similar the learning model is to the actual optimisation space. The second feature affecting the optimisation quality is the size of the particle swarm, which influences optimisation performance by determining the searching ability of the optimisation algorithm.

$$\text{Var.} = [\max(\text{NH}_i) - \min(\text{NH}_i)]/2 \times 100\% \quad (9)$$

$$\text{AI} = \left(\bar{\text{NH}}_{j+1} - \bar{\text{NH}}_j \right) / \bar{\text{NH}}_{j+1} \times 100\% \quad (10)$$

where Var. and AI describe variation and average improvement in optimisation quality, respectively, and i and j describe the particle size and sample size indices, respectively. NH is the average value.

The results illustrated in Figure 11 show that ESO demonstrated distinct improvement in terms of optimisation quality as the size of the sample increased to a satisfactory level, i.e. from 50 to 100 in this case. The variation in optimisation quality also decreased substantially when the sample size reached 100. At this point, the optimisation quality will marginally improve (by less than 5%) as the sample size increases, while variation will stay at a stable level of under 3%. Increasing the sample size from 800 to 4,000 shows average improvement of 0.03%, which indicates that optimisation quality achieves a stable point at a sample size of 800. As for the size of the particle swarm, the results show that this feature can affect the quality of ESO when sample size is held constant. However, the results did not reveal a consistent trend regarding which particle swarm size is most suitable across different sample sizes.

To substantiate the benefits of the proposed approach, the performances of ESO and conventional SO were compared. The developed SO is from AbouRizk and Shi's, (1994)

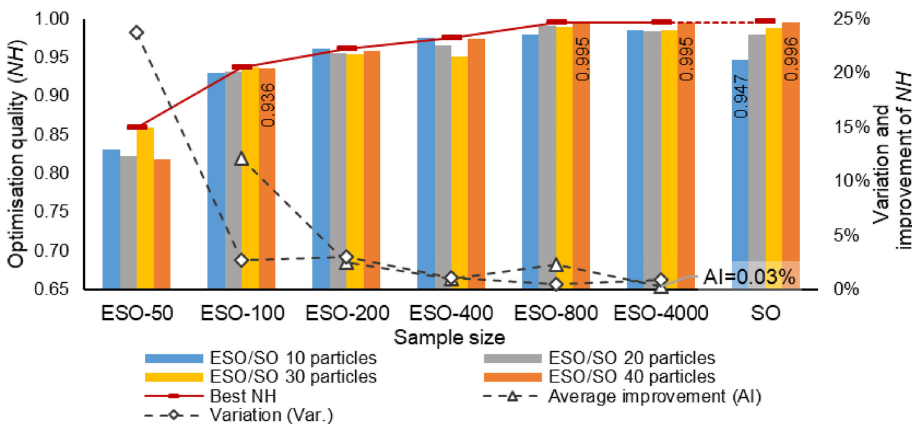


Figure 11. Optimisation performances of ESO and SO with different training and optimisation settings

previous works, and more details can be referred at [Appendix 1](#). The results presented in [Figure 11](#) demonstrate that SO has reduced optimisation quality from 0.996 to 0.947 if the particles are set from 40 to 10. But in this situation, ESO in 400 samples and more has outperformed performance than SO, both in terms of optimisation quality and variation (see [Figure 12](#)). As the sample size grew, i.e. with 800 samples and more, ESO has superior optimisation performance relative to SO with 20 or 30 particles. The ESO with 40 particles is slightly inferior to the SO with the best parameter setting. Therefore, these results demonstrated that ESO outperforms SO when they have a low search ability, while ESO is slightly inferior to SO when they have a high searching ability. These results are consistent with what has been presented in previous research, i.e. a learning model can outperform a SO method in terms of optimisation quality when the learning model and optimisation algorithm are well defined ([Jun et al., 2010](#)).

The key motivation for developing the ESO method was significantly reducing the computational loads of SO and providing efficient optimisation that is practical for use at a construction site. [Figure 13](#) shows that ESO – under satisfactory conditions, i.e. 100 samples

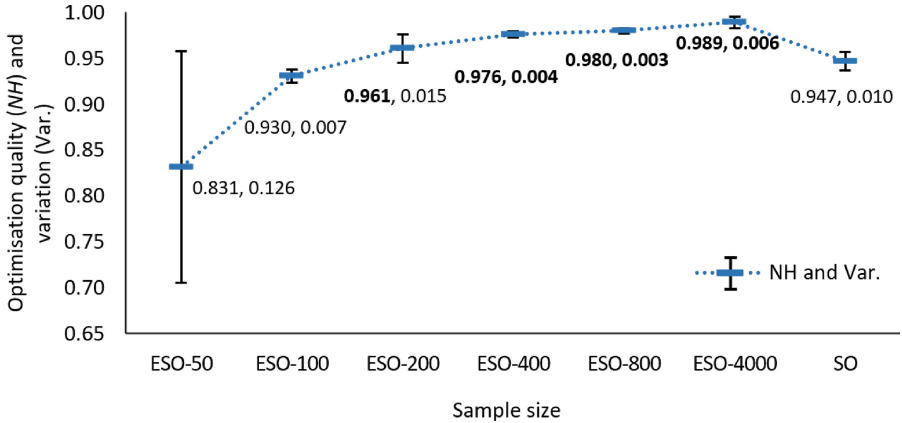


Figure 12. Optimisation performances of ESO and SO with ten particles and different training

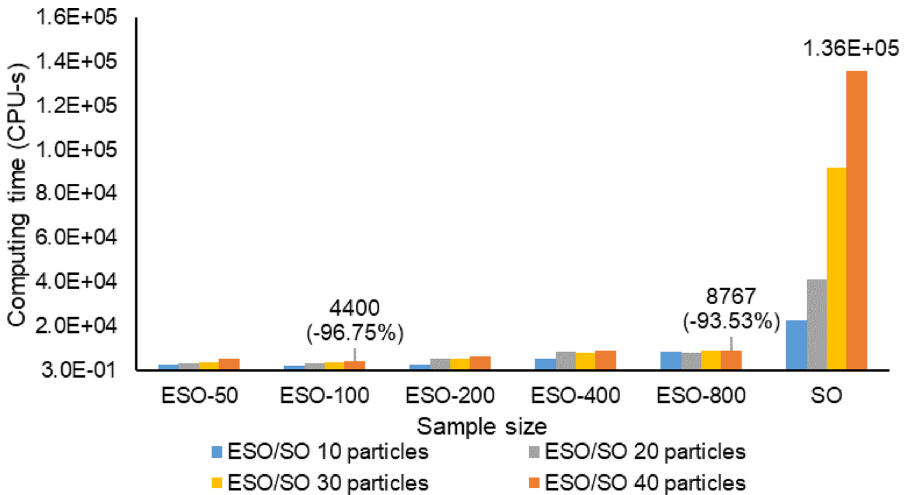


Figure 13. Computing time observed for ESO and SO with different training and optimisation settings

and 40 particles – achieved significant reductions in the computing time associated with the SO method; more specifically, when the particle size was kept constant (both methods had 40 particles), ESO reduced SO computing time by 96.75%. At larger sample sizes, this reduction in computing time was not as significant, yet nevertheless obvious. For example, ESO with 800 samples and 40 particles reduced the computing time of SO under the same settings by 93.53%. However, the computing times of ESO and SO become more comparable at larger sample sizes. For example, performing ESO with 4,000 samples with 40 particles took 3.72×10^4 seconds, which represents a 72.65% reduction in the computational loads of the corresponding SO method. Therefore, applying a large sample size to the training, validation and testing of ESO can partly offset the advantage of computing time reduction. In this case it is important to consider the results depicted in Figure 11, namely, optimisation quality improvements become marginal once the sample size exceeds a satisfactory point. Therefore, applying the ESO approach using satisfied samples is able to noticeably reduce computing time and provides optimisation quality that is comparable to what is observed for the SO method, simultaneously. Sample sizes that exceed the satisfactory point will slightly offset the computing time benefits of ESO without noticeably improving optimisation quality.

To further investigate the influence of sample size on different optimisation performances, the learned optimisation spaces by ESO were compared for different sample sizes. A total of 1,000 random construction samples was used to investigate how the learned optimisation space under different ESO settings compared to the actual optimisation space. The results depicted in Figure 13 show that the learned optimisation space of ESO with 200 samples is far more similar to the actual optimisation space than the learned optimisation space of ESO with 50 samples. In these situations, the ESO will explore the optimisation space in a way that is similar to SO. Theoretically, ESO provides more stable optimisation quality with higher sample size, which was already demonstrated in Figure 11. As the results illustrated in Figure 14 (a, b, c, d) reveal, the learning model successfully interacted with the solution space across all sample sizes. This is an important result, as it demonstrates that the developed algorithm adopts the correct exploitation direction with all sample sizes. ESO trained with more samples will show a learned optimisation space that closely resembles the actual optimization space and, hence, will effectively mimic the conventional SO framework. On the other hand, ESO trained with a limited number of samples will still adopt the correct optimisation direction albeit with a loss in optimisation quality.

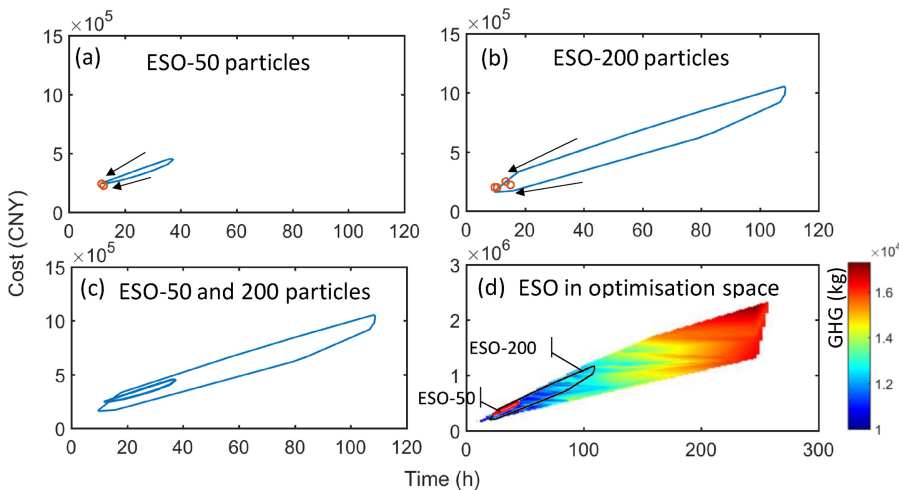


Figure 14.
Optimisation space
learning under
different ESO settings

As the necessary number of optimisation particles should depend on the required searching ability of a specific optimisation problem, more optimisation particles may be required in various applications. Figure 15 shows the optimisation quality and computational loads reduction (relative to SO) of ESO across different numbers of optimisation particles. The presented results show that the computational loads reduction of ESO trained with 100 samples relative to SO is positively related to the number of particles, i.e. progressively increases from 10 to 60 particles. More specifically, ESO reduced the computational loads associated with SO by 91.6%–98.4% as the number of particles increased from 10 to 60. The optimisation quality, on the other hand, of ESO exceeded the minimum value of SO when ESO was trained with 100 samples and 50 particles. ESO trained with 100 samples did not outperform the optimisation quality provided by SO at other particle sizes. However, the largest difference between ESO and SO in terms of performance was only 2.3%; thus, the performances of ESO and SO were comparable when between 10 and 60 particles were applied. To summarise, the computational loads reduction conferred by ESO was more distinct as the increase of required searching ability. Furthermore, neither ESO nor SO showed any consistent superiority in optimisation quality.

As the minimum amount of simulation replication depends on the degree of project uncertainty, more replications are theoretically required for highly uncertain construction projects. A previous construction simulation study (Krantz *et al.*, 2019) reported that an extensive number of replications (50) is required for the optimisation of road construction. Thus, the proposed ESO and SO were also subjected to various amounts of replication to gauge their respective computing time requirements. The number of iterations was set at 100 to minimise the influence of convergence uncertainty, while 100 training samples and 40 particles were used in the models. Figure 16 shows that the ESO approach provides large reductions in computing time as the amount of replications increases. The observed time reduction stabilised (around 90%) when the number of simulation replications surpassed 26, while the net computing time reduction provided by ESO was positively related to the number of simulation replications. Therefore, the results indicate that ESO does not need significantly more computing time when higher simulation replication is required.

Therefore, the proposed ESO has several advantages over SO:

- (1) A well-defined ESO significantly reduces computational loads when compared to SO, yet provides similar optimisation quality (see Figures 11 and 13);
- (2) When the optimisation has a low level of searching ability, ESO outperforms SO both in terms of optimisation quality and variation (see Figure 12);

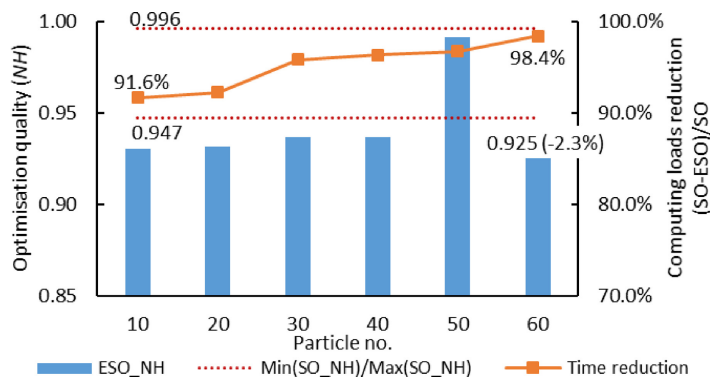


Figure 15. Optimisation quality and computing time of ESO trained with 100 samples relative to SO for various particle swarm sizes

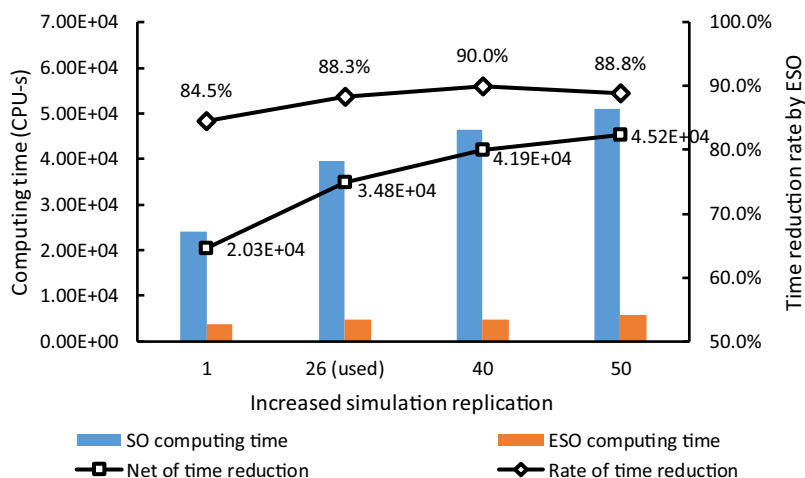


Figure 16. The computing times of ESO and SO when subjected to increased simulation replication

- (3) In situations that require higher searching ability or more simulation replications, ESO can reduce more net computational time compared with SO (see Figures 15 and 16).

6.2 ESO learning method analysis

The chosen learning method is critical for the accuracy of EL. As such, it is necessary to reveal how the training samples and structure of the combination model influence EL accuracy. The performed analyses considered between 50 and 800 training samples, along with combination models with structures ranging from simple to complex, i.e. 120-3-3, 120-10-3, 120-10-10-3 (two hidden layers), respectively. The results in Figure 17 show that the accuracy of the EL model improves as the sample size increases, reaching a stable point at a training sample size of 100. As for the combination model, ESO applying the 120-10-10-3 model showed exhaustive computing time when trained with a small number of samples (e.g. 50). This scenario yielded numerous pseudo solutions, 2,549 to be exact, so the algorithm spent far too much computing time on Pareto solution refinement. Hence, the 120-10-10-3 combination model is impractical for the construction context as the exhaustive identification of a Pareto front offsets the benefits of a developing an approach that is more efficient than conventional SO. The 120-10-3 and 120-3-3 combination models are more practical for decision-makers in the specific case presented in this study. The results revealed that the 120-3-3 network is most suitable in the EL model across sample sizes ranging from 50 to 800. As illustrated in Figure 17, the more complex 120-10-3 structure results in slight overfitting at

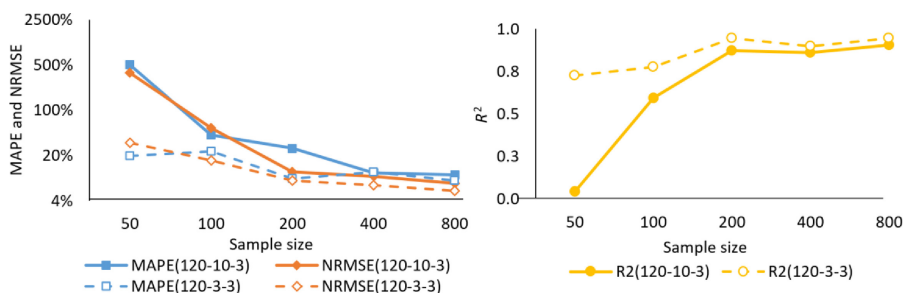


Figure 17. Accuracy of EL at different sample sizes and combination models

small sample sizes, which makes the machine learning model less accurate than the model resulting from the 120-3-3 structure.

In order to measure the performance of ESO by EL model with multiple learners, the performance of the ESO applying EL was compared to methods that applied single learning model, including Bayesian regularization backpropagation neural networks (BRBNN), extreme learning machine (ELM) and regression tree (RT). According to the aforementioned analyses, 100 samples ensures satisfied optimisation quality, efficiency and stable results, while a particle swarm size of 40 provides the best optimisation quality. Thus, these settings were used to test how various learning algorithms affected ESO performance. Table 6 demonstrates that ESO applying EL outperformed the other single learning models in terms of NH. In other words, integrating EL into the optimisation model provides the best optimisation quality. The model that applied ELM demonstrated the shortest training time, which is consistent with previous evidence that ELM is a fast learning algorithm (Huang *et al.*, 2006). This study proposed a machine learning approach to combine multiple learners. The utility of this approach was compared with the commonly applied weighting method, which uses the average weight (AW) for every learner during model combination. Table 6 shows that EL with the proposed machine learning method outperforms the usually applied average weight method in terms of optimisation quality.

The constituent parts of the total running time for the proposed ESO were also revealed. The proposed ESO approach includes three distinct parts of computing time: learning model training, optimisation and Pareto solution refinement. Figure 18 illustrates that learning model training accounts for one of major computing time, with the EL approach having the longest computing time of all the tested learning algorithms. This is reasonable because multiple learners need to be trained in EL.

6.3 Case application and results

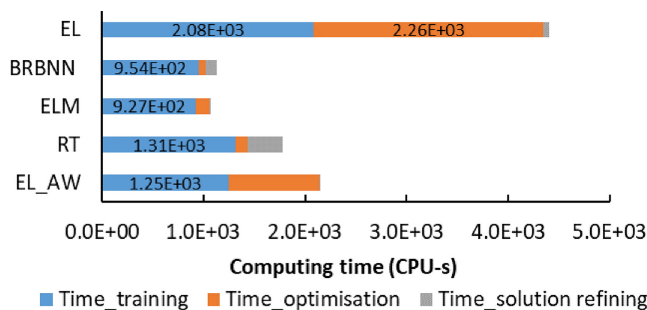
The developed ESO approach was finally used to simulate and optimise the objectives (cost, duration and environmental perspectives) of the case construction project. Based on the

Table 6.
Performances of
different learning
algorithms in ESO

ESO learning method	Optimisation quality (NH)	Total computing time (CPU-s)
EL	<i>0.937</i>	4.40 E+03
BRBNN	0.916	1.08 E+03
ELM	0.608	<i>9.64 E+02</i>
RT	0.854	1.68 E+03
EL_AW	0.878	1.44 E+03

Note(s): Italic numbers indicate the approach with the best performance in the respective aspect; AW represents the average weight method

Figure 18.
Computing time
components of ESO
applying different
learning methods



procedure depicted in Figure 7, the alternative construction scenarios listed in Table 2 were defined in the developed DES simulation model. As in the previous analysis, the number of particles in the PSO algorithm was set to 40, while the EL model was trained with 100 samples. After these steps, the ESO is able to provide an optimised construction solution.

As a result, nine Pareto construction solutions (see Table 7) were obtained after about 1.22 hours of computing time by ESO. In contrast, the SO method with 40 particles included 37.78 hours of computing time. Hence, the on-site manager was provided with a solution (the proposed ESO approach) which yields alternatives that can support decision-making in just over one hour as compared to over day and a half, representing a computing time reduction of 96.75%. As for the optimisation quality, the ESO approach was 6.02%, i.e. $(0.996-0.936)/0.996$, worse than SO in terms of normalised optimisation quality NH.

As can be seen in Table 7, the proposed ESO approach yielded nine non-dominated Pareto solutions (#1~#9) according to the multiple objectives, and they all dominate a random, non-Pareto solution. Pareto solutions #5, #1 and #9 represents the ideal solution for GHG emissions, time and cost, respectively, while the other solutions provide trade-offs between the various objectives. The contractor is then able to use the obtained Pareto solutions to efficiently make a decision.

7. Conclusion

SO is unsuitable to be an on-site real-time decision support tool due to its extensive computational loads. Previous ways that use less searching ability will offset the optimisation quality of decisions. In the application of a large-scale construction project, the SO method took nearly two days to optimise the construction planning characterised by three objectives. The optimisation quality was subsequently reduced when SO was set with fewer particles to reduce the computing time. The present study designs ESO as both computational reduction and quality ensured optimisation approach. The developed ESO was applied to the same project, and the results showed that it can remarkably reduce the computing time associated with SO from two days to one hour while still providing comparable optimisation quality. When the optimisation has 30 or fewer particles, ESO also outperformed SO in terms of optimisation quality both in average level and variation.

In actual construction practices, the dynamic environments in constructions projects should be fulfilled by an efficient optimisation approach for decision-making. ESO can be this approach by its both real-time and reliable optimisation results for supporting decision makings of construction planning. ESO is an alternative to HPC technologies that does not

Scenario	GHG/floor (kg CO ₂ -e)	Duration/floor (day)	Cost/floor (CNY)
Pareto #1	9808.72	8.848	195,057
Pareto #2	8805.62	9.006	188,782
Pareto #3	8789.67	9.167	189,569
Pareto #4	9475.36	8.902	188,277
Pareto #5	8628.37	9.110	197,143
Pareto #6	9141.16	8.875	195,798
Pareto #7	9473.65	8.865	198,978
Pareto #8	8736.81	8.937	194,678
Pareto #9	9746.46	9.065	187,970
A non-Pareto solution	10649.50	15.875	248,205
GHG best scenario	Pareto #5	-	-
Duration best scenario	-	Pareto #1	-
Cost best scenario	-	-	Pareto #9

Table 7.
Performances of nine
Pareto solutions and a
random non-Pareto
solution

require extensive computational capacity and can be easily used in actual on-site environments. The proposed method in this paper retains the simulation and optimisation sections of SO, yet embeds an ensemble learning algorithm within the conventional SO framework. As a result, the ESO approach can be run on a standard desktop and laptop, providing optimisation results in a few hours.

Furthermore, ESO can reduce more computational loads associated with SO especially in two situations. The one is the projects inherently having high levels of uncertainty, so they require a substantial number of simulation replications to provide reliable optimisations. The other is the projects have a wide range of construction plan scenarios, i.e. have wide optimisation spaces, so they require high level of searching ability in the optimisation model. The presented ESO showed more computational loads reductions compared with SO when either the number of simulation replications or particles grew.

The proposed ESO is a promising tool for efficient construction optimisation, but it still has certain limitations need to be addressed in further work. Simulation model maintenance presents a problem in the construction setting which could reduce the utility of ESO especially with the progress of construction. Inspired by the recent study of [Shrestha and Behzadan \(2018\)](#), the application of a real-time data collection technology (e.g. Internet of things) could be a promising solution to overcome this problem. Then, the proposed ESO that connected with real-time data could enable automatic or semi-automatic construction optimisation. In addition, although this study verified the benefits of ESO in a large-scale building construction, it is also interesting to test present work in a civil engineering such as bridges or roads construction due to their inherent large scales, wider range of scenarios and long-term duration.

References

- AbouRizk, S. and Shi, J. (1994), "Automated construction-simulation optimization", *Journal of Construction Engineering and Management*, Vol. 120 No. 2, pp. 374-385, doi: [10.1061/\(ASCE\)0733-9364\(1994\)120:2\(374\)](#).
- Adeli, H. (2001), "Neural networks in civil engineering: 1989-2000", *Computer-Aided Civil and Infrastructure Engineering*, Vol. 16 No. 2, pp. 126-142, doi: [10.1111/0885-9507.00219](#).
- Adeli, H. and Karim, A. (1997), "Scheduling/cost optimization and neural dynamics model for construction", *Journal of Construction Engineering and Management*, Vol. 123 No. 4, pp. 450-458, doi: [10.1061/\(ASCE\)0733-9364\(1997\)123:4\(450\)](#).
- Al Hattab, M., Zankoul, E. and Hamzeh, F.R. (2017), "Near-real-time optimization of overlapping tower crane operations: a model and case study", *Journal of Computing in Civil Engineering*, Vol. 31 No. 4, p. 05017001, doi: [10.1061/\(ASCE\)CP.1943-5487.0000666](#).
- Apostu, A., Puican, F., Ularu, G., Suci, G. and Todoran, G. (2013), "Study on advantages and disadvantages of cloud computing—the advantages of telemetry applications in the cloud", *Proceedings of the 13th International Conference on Applied Computer Science*, Morioka City, Iwate, Japan, pp. 118-123, ISBN: 978-1-61804-179-1.
- Banks, J. (1998), *Handbook of Simulation: Principles, Methodology, Advances, Applications, and Practice*, John Wiley & Sons, Hoboken, NJ, ISBN: 978-0-471-13403-9.
- Bratton, D. and Kennedy, J. (2007), "Defining a standard for particle swarm optimization", *2007 IEEE Swarm Intelligence Symposium*, Honolulu, HI, USA, pp. 120-127, doi:ISBN: 978-1-4244-0708-8.
- Breiman, L. (1996), "Bagging predictors", *Machine Learning*, Vol. 24 No. 2, pp. 123-140, doi: [10.1023/A:1018054314350](#).
- Breiman, L., Friedman, J., Stone, C.J. and Olshen, R.A. (1984), *Classification and Regression Trees*, Chapman and Hall/CRC, New York, NY, ISBN: 978-0412048418.
- Budgaga, W., Malensek, M., Pallickara, S., Harvey, N., Breidt, F.J. and Pallickara, S. (2016), "Predictive analytics using statistical, learning, and ensemble methods to support real-time exploration of

- discrete event simulations”, *Future Generation Computer Systems*, Vol. 56, pp. 360-374, doi: [10.1016/j.future.2015.06.013](https://doi.org/10.1016/j.future.2015.06.013).
- Cao, Y., Ashuri, B. and Baek, M. (2018), “Prediction of unit price bids of resurfacing highway projects through ensemble machine learning”, *Journal of Computing in Civil Engineering*, Vol. 32 No. 5, p. 04018043, doi: [10.1061/\(ASCE\)CP.1943-5487.0000788](https://doi.org/10.1061/(ASCE)CP.1943-5487.0000788).
- Cassandras, C.G. and Lafortune, S. (2009), *Introduction to Discrete Event Systems*, Springer, Boston, MA, ISBN: 978-0-387-33332-8.
- Cheng, T.-M. and Feng, C.-W. (2003), “An effective simulation mechanism for construction operations”, *Automation in Construction*, Vol. 12 No. 3, pp. 227-244, doi: [10.1016/S0926-5805\(02\)00086-9](https://doi.org/10.1016/S0926-5805(02)00086-9).
- Colak, T. and Qahwaji, R. (2009), “Automated solar activity prediction: a hybrid computer platform using machine learning and solar imaging for automated prediction of solar flares”, *Space Weather*, Vol. 7, p. S06001, doi: [10.1029/2008sw000401](https://doi.org/10.1029/2008sw000401).
- Collyer, S. and Warren, C.M. (2009), “Project management approaches for dynamic environments”, *International Journal of Project Management*, Vol. 27 No. 4, pp. 355-364, doi: [10.1016/j.ijproman.2008.04.004](https://doi.org/10.1016/j.ijproman.2008.04.004).
- Eglajs, V. and Audze, P. (1977), “New approach to the design of multifactor experiments (in Russian)”, *Problems of Dynamics and Strengths*, Vol. 35 No. 1, pp. 104-107.
- Erdal, H. and Karahanoglu, I. (2016), “Bagging ensemble models for bank profitability: an empirical research on Turkish development and investment banks”, *Applied Soft Computing*, Vol. 49, pp. 861-867, doi: [10.1016/j.asoc.2016.09.010](https://doi.org/10.1016/j.asoc.2016.09.010).
- Eshtehardian, E., Afshar, A. and Abbasnia, R. (2009), “Fuzzy-based MOGA approach to stochastic time-cost trade-off problem”, *Automation in Construction*, Vol. 18 No. 5, pp. 692-701, doi: [10.1016/j.autcon.2009.02.001](https://doi.org/10.1016/j.autcon.2009.02.001).
- Feng, K. (2019), “Research on construction process environmental simulation and optimization under deep uncertainty (in Chinese)”, *Department of Construction Management*, Harbin, Harbin Institute of Technology, p. 146.
- Feng, K., Lu, W., Chen, S. and Wang, Y. (2018), “An integrated environment–cost–time optimisation method for construction contractors considering global warming”, *Sustainability*, Vol. 10 No. 11, p. 4207, doi: [10.3390/su10114207](https://doi.org/10.3390/su10114207).
- Feng, K., Lu, W. and Wang, Y. (2019), “Assessing environmental performance in early building design stage: an integrated parametric design and machine learning method”, *Sustainable Cities and Society*, Vol. 50, p. 101596, doi: [10.1016/j.scs.2019.101596](https://doi.org/10.1016/j.scs.2019.101596).
- Foresee, F.D. and Hagan, M.T. (1997), “Gauss-Newton approximation to Bayesian learning”, *Proceedings of International Conference on Neural Networks (ICNN'97)*, Houston, TX, USA, pp. 1930-1935, ISBN: 0-7803-4122-8.
- Gilliland, M. (2010), *The Business Forecasting Deal: Exposing Myths, Eliminating Bad Practices, Providing Practical Solutions*, John Wiley & Sons, Hoboken, NJ, ISBN: 9780470574430.
- Gosavi, A. (2015), *Simulation-Based Optimization*, Springer, New York, NY, ISBN: 978-1-4899-7490-7.
- Hall, V., Sklepari, M. and Rodger, A. (2014), “Protein secondary structure prediction from circular dichroism spectra using a self-organizing map with concentration correction”, *Chirality*, Vol. 26 No. 9, pp. 471-482, doi: [10.1002/chir.22338](https://doi.org/10.1002/chir.22338).
- Htike, K.K. (2016), “Efficient determination of the number of weak learners in AdaBoost”, *Journal of Experimental and Theoretical Artificial Intelligence*, Vol. 29 No. 5, pp. 1-16, doi: [10.1080/0952813X.2016.1266038](https://doi.org/10.1080/0952813X.2016.1266038).
- Huang, G.B., Zhu, Q.Y. and Siew, C.K. (2006), “Extreme learning machine: theory and applications”, *Neurocomputing*, Vol. 70 Nos 1-3, pp. 489-501, doi: [10.1016/j.neucom.2005.12.126](https://doi.org/10.1016/j.neucom.2005.12.126).
- João, M.-M., Soares, C. and Sousa, J.F.D. (2012), “Ensemble approaches for regression: a survey”, *ACM Computing Surveys*, Vol. 45 No. 1, pp. 1-40, doi: [10.1145/2379776.2379786](https://doi.org/10.1145/2379776.2379786).

- Jun, Z., Yu-An, T., Xue-Lan, Z. and Jun, L. (2010), "An improved dynamic structure-based neural networks determination approaches to simulation optimization problems", *Neural Computing and Applications*, Vol. 19 No. 6, pp. 883-901, doi: [10.1007/s00521-010-0348-x](https://doi.org/10.1007/s00521-010-0348-x).
- Kamali, M. and Hewage, K. (2017), "Development of performance criteria for sustainability evaluation of modular versus conventional construction methods", *Journal of Cleaner Production*, Vol. 142, pp. 3592-3606, doi: [10.1016/j.jclepro.2016.10.108](https://doi.org/10.1016/j.jclepro.2016.10.108).
- Kennedy, J. and Eberhart, R. (1995), "Particle swarm optimization", *1995 IEEE International Conference on Neural Networks Proceedings*, New York, USA, pp. 1942-1948, ISBN: 0-7803-2768-3.
- Kennedy, J., Eberhart, R.C. and Shi, Y. (2006), *Swarm Intelligence*, Morgan Kaufmann Publishers, San Francisco, ISBN: 1-55860-595-9.
- Kim, H.C., Pang, S., Je, H.M., Kim, D. and Bang, S.Y. (2003), "Constructing support vector machine ensemble", *Pattern Recognition*, Vol. 3612, pp. 2757-2767, doi: [10.1016/s0031-3203\(03\)00175-4](https://doi.org/10.1016/s0031-3203(03)00175-4).
- Krantz, J., Feng, K.L., Larsson, J. and Olofsson, T. (2019), "'Eco-Hauling' principles to reduce carbon emissions and the costs of earthmoving - a case study", *Journal of Cleaner Production*, Vol. 208, pp. 479-489, doi: [10.1016/j.jclepro.2018.10.113](https://doi.org/10.1016/j.jclepro.2018.10.113).
- Krogh, A. and Vedelsby, J. (1995), "Neural network ensembles, cross validation, and active learning", *Proceedings of the 8th International Conference on Neural Information Processing Systems*, pp. 231-238, ISBN: 0262201046.
- Larsson, J., Lu, W., Krantz, J. and Olofsson, T. (2016), "Discrete event simulation analysis of product and process platforms: a bridge construction case study", *Journal of Construction Engineering and Management*, Vol. 142 No. 4, p. 04015097, doi: [10.1061/\(ASCE\)CO.1943-7862.0001093](https://doi.org/10.1061/(ASCE)CO.1943-7862.0001093).
- Lee, J.-S., Filatova, T., Ligmann-Zielinska, A., Hassani-Mahmooui, B., Stonedahl, F., Lorscheid, I., Voinov, A., Polhill, J.G., Sun, Z. and Parker, D.C. (2015), "The complexities of agent-based modeling output analysis", *Journal of Artificial Societies and Social Simulation*, Vol. 18 No. 4, p. 4, doi: [10.18564/jasss.2897](https://doi.org/10.18564/jasss.2897).
- Liao, C.-W. (2012), "Analysis of occupational accidents during construction of buildings using classification and regression tree", *Instrumentation, Measurement, Circuits and Systems, AISC 127*, Berlin, Heidelberg, pp. 1003-1010, ISBN: 978-3-642-27334-6.
- Lorscheid, I., Heine, B.-O. and Meyer, M. (2012), "Opening the 'black box' of simulations: increased transparency and effective communication through the systematic design of experiments", *Computational and Mathematical Organization Theory*, Vol. 18 No. 1, pp. 22-62, doi: [10.1007/s10588-011-9097-3](https://doi.org/10.1007/s10588-011-9097-3).
- Love, P.E., Holt, G.D., Shen, L.Y., Li, H. and Irani, Z. (2002), "Using systems dynamics to better understand change and rework in construction project management systems", *International Journal of Project Management*, Vol. 20 No. 6, pp. 425-436, doi: [10.1016/S0263-7863\(01\)00039-4](https://doi.org/10.1016/S0263-7863(01)00039-4).
- Lu, W. and Olofsson, T. (2014), "Building information modeling and discrete event simulation: towards an integrated framework", *Automation in Construction*, Vol. 44, pp. 73-83, doi: [10.1016/j.autcon.2014.04.001](https://doi.org/10.1016/j.autcon.2014.04.001).
- Magnier, L. and Haghghat, F. (2010), "Multiobjective optimization of building design using TRNSYS simulations, genetic algorithm, and artificial neural network", *Building and Environment*, Vol. 45 No. 3, pp. 739-746, doi: [10.1016/j.buildenv.2009.08.016](https://doi.org/10.1016/j.buildenv.2009.08.016).
- Mao, C., Shen, Q., Shen, L. and Tang, L. (2013a), "Comparative study of greenhouse gas emissions between off-site prefabrication and conventional construction methods: two case studies of residential projects", *Energy and Buildings*, Vol. 66 No. 5, pp. 165-176, doi: [10.1016/j.enbuild.2013.07.033](https://doi.org/10.1016/j.enbuild.2013.07.033).
- Mao, C., Shen, Q.P., Shen, L.Y. and Tang, L.Y.N. (2013b), "Comparative study of greenhouse gas emissions between off-site prefabrication and conventional construction methods: two case studies of residential projects", *Energy and Buildings*, Vol. 66, pp. 165-176, doi: [10.1016/j.enbuild.2013.07.033](https://doi.org/10.1016/j.enbuild.2013.07.033).

- Nakayama, H., Arakawa, M. and Sasaki, R. (2002), "Simulation-based optimization using computational intelligence", *Optimization and Engineering*, Vol. 3 No. 2, pp. 201-214, doi: [10.1023/A:1020971504868](https://doi.org/10.1023/A:1020971504868).
- NDRC (2011), *Provincial Greenhouse Gas Inventory Guidelines*, National Development and Reform Commission, Beijing.
- Nguyen, A.T., Reiter, S. and Rigo, P. (2014), "A review on simulation-based optimization methods applied to building performance analysis", *Applied Energy*, Vol. 113, pp. 1043-1058, doi: [10.1016/j.apenergy.2013.08.061](https://doi.org/10.1016/j.apenergy.2013.08.061).
- Ninić, J. and Meschke, G. (2015), "Model update and real-time steering of tunnel boring machines using simulation-based meta models", *Tunnelling and Underground Space Technology*, Vol. 45, pp. 138-152, doi: [10.1016/j.tust.2014.09.013](https://doi.org/10.1016/j.tust.2014.09.013).
- Rohler, A.B. and Chen, S. (2011), "An analysis of sub-swarms in multi-swarm systems", in Wang, D.H. and Reynolds, M. (Eds), *AI 2011: Advances in Artificial Intelligence*, Vol. 7106, pp. 271-280, ISBN: 978-3-642-25831-2.
- Rumelhart, D.E., Hinton, G.E. and Williams, R.J. (1985), *Learning Internal Representations by Error Propagation*, MIT Press, Cambridge, ISBN: 9780262291408.
- Salimi, S., Mawlana, M. and Hammad, A. (2018), "Performance analysis of simulation-based optimization of construction projects using high performance computing", *Automation in Construction*, Vol. 87, pp. 158-172, doi: [10.1016/j.autcon.2017.12.003](https://doi.org/10.1016/j.autcon.2017.12.003).
- SCPCM (2016a), *2016 Shenzhen Consumption Quota of Construction Engineering, Shenzhen_Construction_Project_Cost_Management_Station*, China Building Industry Press, Shenzhen, ISBN: 9787112205233.
- SCPCM (2016b), *2016 Shenzhen Consumption Quota of Prefabricated Building Projects Shenzhen_Construction_Project_Cost_Management_Station*, China Building Industry Press, Shenzhen, ISBN: 9787112197279.
- Segerstedt, A. and Olofsson, T. (2010), "Supply chains in the construction industry", *Supply Chain Management: An International Journal*, Vol. 15 No. 5, pp. 347-353, doi: [10.1108/13598541011068260](https://doi.org/10.1108/13598541011068260).
- Shin, Y. (2015), "Application of boosting regression trees to preliminary cost estimation in building construction projects", *Computational Intelligence and Neuroscience*, Vol. 2015, p. 149702, doi: [10.1155/2015/149702](https://doi.org/10.1155/2015/149702).
- Shin, Y., Cho, H. and Kang, K.-I. (2011), "Simulation model incorporating genetic algorithms for optimal temporary hoist planning in high-rise building construction", *Automation in Construction*, Vol. 20 No. 5, pp. 550-558, doi: [10.1016/j.autcon.2010.11.021](https://doi.org/10.1016/j.autcon.2010.11.021).
- Shrestha, P. and Behzadan, A.H. (2018), "Chaos theory-inspired evolutionary method to refine imperfect sensor data for data-driven construction simulation", *Journal of Construction Engineering and Management*, Vol. 144 No. 3, p. 04018001, doi: [10.1061/\(ASCE\)CO.1943-7862.0001441](https://doi.org/10.1061/(ASCE)CO.1943-7862.0001441).
- Song, M.P. and Gu, G.C. (2004), "Research on particle swarm optimization: a review", *Proceedings of 2004 International Conference on Machine Learning and Cybernetics*, Vol. 4, pp. 2236-2241, doi: [10.1109/ICMLC.2004.1382171](https://doi.org/10.1109/ICMLC.2004.1382171).
- Tavakolan, M., Mohammadi, S. and Zahraie, B. (2019), "Construction and resource short-term planning using a BIM-based ontological decision support system", *Canadian Journal of Civil Engineering*, Vol. 48 No. 1, pp. 75-88, doi: [10.1139/cjce-2019-0439](https://doi.org/10.1139/cjce-2019-0439).
- Wang, Y.R., Yu, C.Y. and Chan, H.H. (2012), "Predicting construction cost and schedule success using artificial neural networks ensemble and support vector machines classification models", *International Journal of Project Management*, Vol. 30 No. 4, pp. 470-478, doi: [10.1016/j.ijproman.2011.09.002](https://doi.org/10.1016/j.ijproman.2011.09.002).
- Wang, Y., Feng, K. and Lu, W. (2017), "An environmental assessment and optimization method for contractors", *Journal of Cleaner Production*, Vol. 142, pp. 1877-1891, doi: [10.1016/j.jclepro.2016.11.097](https://doi.org/10.1016/j.jclepro.2016.11.097).

- Webb, G.I. and Zheng, Z.J. (2004), "Multistrategy ensemble learning: reducing error by combining ensemble learning techniques", *IEEE Transactions on Knowledge and Data Engineering*, Vol. 16 No. 8, pp. 980-991, doi: [10.1109/tkde.2004.29](https://doi.org/10.1109/tkde.2004.29).
- Yan, Z. and Wang, J. (2014), "Robust model predictive control of nonlinear systems with unmodeled dynamics and bounded uncertainties based on neural networks", *IEEE Transactions on Neural Networks and Learning Systems*, Vol. 25 No. 3, pp. 457-469, doi: [10.1109/TNNLS.2013.2275948](https://doi.org/10.1109/TNNLS.2013.2275948).
- Yang, S. and Browne, A. (2004), "Neural network ensembles: combining multiple models for enhanced performance using a multistage approach", *Expert Systems*, Vol. 21 No. 5, pp. 279-288, doi: [10.1111/j.1468-0394.2004.00285.x](https://doi.org/10.1111/j.1468-0394.2004.00285.x).
- Yang, I.-T., Hsieh, Y.-M. and Kung, L.-O. (2012), "Parallel computing platform for multiobjective simulation optimization of bridge maintenance planning", *Journal of Construction Engineering and Management*, Vol. 138 No. 2, pp. 215-226, doi: [10.1061/\(ASCE\)CO.1943-7862.0000421](https://doi.org/10.1061/(ASCE)CO.1943-7862.0000421).
- Yu, W., Lo, S. and Fan, G. (2005), "Real-time decision-making with partial information for construction management", *2005 Proceedings of the 22nd ISARC*, Ferrara, Italy, September 11-14, 2005, doi: [10.22260/ISARC2005/0088](https://doi.org/10.22260/ISARC2005/0088).
- Zhang, H. (2008), "Multi-objective simulation-optimization for earthmoving operations", *Automation in Construction*, Vol. 18 No. 1, pp. 79-86, doi: [10.1016/j.autcon.2008.05.002](https://doi.org/10.1016/j.autcon.2008.05.002).
- Zhang, H. and Li, H. (2004), "Simulation-based optimization for dynamic resource allocation", *Automation in Construction*, Vol. 13 No. 3, pp. 409-420, doi: [10.1016/j.autcon.2003.12.005](https://doi.org/10.1016/j.autcon.2003.12.005).
- Zhang, C. and Ma, Y. (2012), *Ensemble Machine Learning: Methods and Applications*, Springer, ISBN: 978-1-4419-9326-7.
- Zhang, Y.S. and Ng, S.T. (2012), "An ant colony system based decision support system for construction time-cost optimization", *Journal of Civil Engineering and Management*, Vol. 18 No. 4, pp. 580-589, doi: [10.3846/13923730.2012.704164](https://doi.org/10.3846/13923730.2012.704164).
- Zhou, Z.-H. (2009), "Ensemble learning", *Encyclopedia of Biometrics*, No. 1, pp. 270-273, doi: [10.1007/978-0-387-73003-5_293](https://doi.org/10.1007/978-0-387-73003-5_293).
- Zitzler, E., Thiele, L., Laumanns, M., Fonseca, C.M. and da Fonseca, V.G. (2003), "Performance assessment of multiobjective optimizers: an analysis and review", *IEEE Transactions on Evolutionary Computation*, Vol. 7 No. 2, pp. 117-132, doi: [10.1109/tevc.2003.810758](https://doi.org/10.1109/tevc.2003.810758).

A1: ENSEMBLE LEARNING EMBEDDED SIMULATION OPTIMISATION

Construction dataset generation

```

while sample  $i$ -th < required sample size  $I$ , do
  LHS sampling:  $X_i = \{x_1 \sim [x_{1\min}, x_{1\max}], x_2 \sim [x_{2\min}, x_{2\max}], \dots\}$ 
  divide parameters' values into equal probability regions  $[x_{\min}, x_{\max}] = [(x_{\min}, x_2),$ 
 $(x_2, x_3), \dots, x_{\max}]$ 
  sampling regions  $(x_i, x_{i+1})$ 
  sampling value in regions  $x_i \sim (x_i, x_{i+1})$ 
  input  $X_i$  ( $i=1, \dots, I$ ) into DES model
  run DES model
   $i \leftarrow i+1$ 
return  $S = \{S_{\text{train}}, S_{\text{test}}, S_{\text{validate}}\}$ 
end

```

EL model establishment

```

Bagging  $S_m \sim S_{\text{train}}$  ( $m=1, 2, \dots, M$ )
while learner  $f_m$   $m$ -th < learner size  $M$ , do
  train learner  $f_m$  with  $S_m$ 
   $m \leftarrow m+1$ 
end
train combination model with  $S_{\text{test}}$ 
input the outputs of  $f_m$  and use outputs in actual performances in  $S_{\text{test}}$ 
 $F \leftarrow$  combination model
validate EL model with  $S_{\text{validate}}$ 
return  $R^2$ , NRMSE, MAPE
if Not accepted
  do Construction dataset generation
  do EL model establishment
else
  finish EL model validate
end

```

Learning-based optimisation

```

initialisation iteration  $t=1$ , particle position  $P$ 
  PSO global best record  $gbestP$ , PSO individual best record  $pbestP$ 
while optimisation criteria:  $t <$  maximum iteration  $T$  || optimisation convergence
  condition || replicated mechanism  $n <$  maximum replication  $N$  do
  PSO searching:
   $V(t) \leftarrow w(t) \times V(t-1) + c_1 \times r_1 \times (pbestP(t-1) - P(t-1)) + c_2 \times r_2 \times (gbestP(t-1) - P(t-1))$  &&
   $P(t) \leftarrow P(t-1) + V(t)$ 
  while particle  $k$ -th < particle size  $K$ , do
    run EL model to evaluate particle  $k$ -th
    return particle  $k$ -th fitness in "cost-time-GHG"
     $k \leftarrow k+1$ 
  end
  update  $gbestP$  and  $pbestP$ 
   $t \leftarrow t+1$ 
end
return pseudo solutions

```

Pareto solution refinement

```

Input pseudo solutions into DES model
run DES model
return "cost-time-GHG"
remove solutions being dominated
return timely Pareto solutions

```

```

A2: SIMULATION-BASED OPTIMISATION

initialisation iteration  $t=1$ , particle position  $P$ 
                PSO global best record  $gbestP$ , PSO individual best record  $pbestP$ 
while optimisation criteria:  $t < \text{maximum iteration } T$  || optimisation convergence
condition || replicated mechanism  $n < \text{maximum replication } N$  do
  PSO searching:
   $V(t) \leftarrow w(t) \times V(t-1) + c_1 \times r_1 \times (pbestP(t-1) - P(t-1)) + c_2 \times r_2 \times (gbestP(t-1) - P(t-1))$  &&
   $P(t) \leftarrow P(t-1) + V(t)$ 
  while particle  $k$ -th  $<$  particle size  $K$ , do
    run DES model directly for position of  $k$ -th particle
    return particle  $k$ -th fitness in "cost-time-GHG"
     $k \leftarrow k+1$ 
  end
  update  $gbestP$  and  $pbestP$ 
   $t \leftarrow t+1$ 
end
return Pareto solutions
    
```

Appendix 2
DES model validation

Testing item	Unit	Document quantities	Simulation
Cost of cranes	CNY	965463.344	1,089,140
Cost of construction lifts	CNY	1405916.4	1,405,920
Cost of concrete pumps	CNY	502863.2	502,863
Labour for CS beam and slab rebar process	CNY	185584.042	185,584
Labour for CS wall and column rebar process	CNY	148282.393	148,282
Labour for PC external wall hoist and installation	CNY	1175755.03	1,175,760
Labour for PC balcony hoist and installation	CNY	91835.8134	91835.8
Labour for PC internal wall hoist and installation	CNY	292757.501	292,758
Labour for CS concrete pump, vibration and curving	CNY	997323.92	997,324
Labour for PC slab hoist and installation	CNY	412919.246	412,919
Labour for PC beam hoist and installation	CNY	140441.223	140,441
Labour for PC stair hoist and installation	CNY	123976.842	123,977
Labour for CS wall and column rebar installation	CNY	310161.014	310,161
Labour for joint grout	CNY	614074.102	614,074
Labour for climbing formwork	CNY	3,024,810	3,024,810
Diesel consumption of PC trucks	kg	50775.386	50775.4
Power consumption of concrete pumps	kWh	15025.5102	15,747
Power consumption of cranes	kWh	486,623	491,892
Power consumption of construction lifts	kWh	75113.775	75113.8

Table A1.
The outcome of the simulation and the construction documents of a typical floor

Null hypothesis	Test	Sig	Decision
There is no difference between the median values for the real construction data and the simulation	Wilcoxon signed-rank test	0.231	Retain the null hypothesis

Note(s): The nonparametric Wilcoxon signed-rank test was used when comparing the simulation and the documents (see Table A1). The resulting statistic for the test was 0.231, which exceeds the 0.050 significance level (see Table A2), and it means that the simulated and real data do not statistically differ at the 95% confidence level

Table A2.
Wilcoxon signed-rank test results

Author Affiliations

Kailun Feng, Department of Construction Management, Key Lab of Structures Dynamic Behavior and Control of the Ministry of Education, Harbin Institute of Technology, Harbin, China; Key Lab of Smart Prevention and Mitigation of Civil Engineering Disasters of the Ministry of Industry and Information Technology, Harbin Institute of Technology, Harbin, China and Department of Civil, Environmental, and Natural Resources Engineering, Luleå University of Technology, Luleå, Sweden

Shiwei Chen, Department of Civil, Environmental, and Natural Resources Engineering, Luleå University of Technology, Luleå, Sweden

Weizhuo Lu, Department of Applied Physics and Electronics, Umeå University, Umeå, Sweden and Department of Civil, Environmental, and Natural Resources Engineering, Luleå University of Technology, Luleå, Sweden

Shuo Wang, Department of Construction Management, Key Lab of Structures Dynamic Behavior and Control of the Ministry of Education, Harbin Institute of Technology, Harbin, China and Key Lab of Smart Prevention and Mitigation of Civil Engineering Disasters of the Ministry of Industry and Information Technology, Harbin Institute of Technology, Harbin, China

Bin Yang, School of Energy and Safety Engineering, Tianjin Chengjian University, Tianjin, China and Department of Applied Physics and Electronics, Umeå University, Umeå, Sweden

Chengshuang Sun, School of Urban Economics and Management, Beijing University of Civil Engineering and Architecture, Beijing, China

Yaowu Wang, Department of Construction Management, Key Lab of Structures Dynamic Behavior and Control of the Ministry of Education, Harbin Institute of Technology, Harbin, China and Key Lab of Smart Prevention and Mitigation of Civil Engineering Disasters of the Ministry of Industry and Information Technology, Harbin Institute of Technology, Harbin, China

Corresponding author

Weizhuo Lu can be contacted at: weizhuo.lu@umu.se