Optimized Production Scheduling for Modular Construction Manufacturing

A. Bhatia[a], O. Moselhi[a,b] and S.H. Han[a,b]

[a] Department of Building, Civil and Environmental Engineering, Concordia University, Montreal, QC, Canada
[b] Centre for Innovation in Construction and Infrastructure Engineering and Management (CICIEM), Concordia University, Montreal, QC, Canada
angatpalsingh.bhatia@concordia.ca, moselhi@encs.concordia.ca, sanghyeok.han@concordia.ca

Abstract –

Modular construction is becoming a viable construction method in North America due to its support of the concept of circular construction and its inherent ability to provide a faster return on investment. The process of modular construction manufacturing (MCM) operates as a production line, where the number of module components (e.g., wall, roof, and floor panels) with different design specifications and their sequence (i.e., order of prefabricating these module components) dictates the productivity of the production line. This variation in design specifications and impractical sequences of module components leads to imbalanced production lines and prolonged makespan (i.e., total completion time) of prefabricating module components at workstations. To address these challenges, this paper proposes a method that utilizes deep neural network and genetic algorithm (GA) techniques to solve the modular construction manufacturing scheduling problem (MCMSP). The method consists of two processes: *(i)* developing a deep neural network model based on the historical time data and later hyperparameter tuning using a GA in order to select the optimal neural network configurations*;* and *(ii)* subsequently using the predicted process times as input in the optimization model in order to schedule the sequences of module components (e.g., wall panels). The proposed method is implemented in a wood-based wall panel production line of a modular fabricator in Edmonton, Canada. This developed method can assist production managers in efficiently forecasting process times and developing production line schedules.

Keywords –

Modular Construction Manufacturing (MCM); Production Line; Deep Neural Network; Scheduling; Optimization

1. Introduction and Background

Modular construction is a process where module components (e.g., wall, roof, and floor panels) are prefabricated in a controlled factory environment and later transported on-site to be installed as building blocks. Modular construction is increasingly growing in popularity over conventional construction, given its potential to achieve shortened construction schedules and less material waste, facilitate the process of circular construction and lead to a quicker return on investment for project owners [1] and [2]. However, prefabricating modules has been a very complex manufacturing process due to the highly customized nature of module components (i.e., variation in design specifications), leading to varying production rates and imbalanced production lines. Therefore, this process makes it challenging for production line managers to develop a robust production planning and scheduling model without accurately forecasting the process times of module components at workstations along the production line.

In modular construction manufacturing (MCM), the processing time is defined as the time taken (i.e., start and finish time) to complete one module component (e.g., wall panel) at the workstation [3]. Accurately predicting the process times can be seen as a way of: *(i)* gaining an in-depth understanding of the nature of module components; and *(ii)* making data-driven decisions with respect to the planning and scheduling of module components in the production line. However, the modular construction industry has not achieved its full potential benefits due to the lack of a systematic, data-driven decision-making approach in order to solve the modular construction manufacturing scheduling problem (MCMSP). In practice, production managers make guesswork based on their experience and rely on the average process times. However, such methods do not provide optimal results due to the stochastic nature (i.e., randomness) of prefabricating module components at workstations. In this respect, this paper proposes a method combining deep neural networks and GA to develop a model to solve MCMSP efficiently. The proposed method encompasses; *(i)* data preprocessing (e.g., outlier's removal) in order to clean RFID-based production line data for prediction purposes; *(ii)* development of predictive models using the deep neural network; *(iii)* identification of optimal model configuration for the deep neural network using GA optimization; and *(iv)* development of a scheduling model using GA based optimization technique facilitating the data-driven production line schedules.

2. Literature Review

This section provides a literature review from three perspectives: *(i)* Data analytics; *(ii)* Scheduling in MCM; and *(iii)* optimization for production scheduling.

2.1 Data Analytics

Various machine learning and statistical techniques were employed to substantially enhance the accuracy of predictive models in construction and infrastructure engineering [4]. Artificial neural networks (ANN) are a more suitable tool for developing predictive models as they provide superior performance for highly uncertain, nonlinear, and complicated problems [5]. According to Moselhi et al. [6], ANN encompasses a collection of processing elements, usually organized into layers (i.e., input, hidden, and output layers). The input layer accepts the data (i.e., independent variables), which is used by the hidden layers to represent the relationship and the output layer produces the network response (i.e., dependent variable). To make reliable predictions, ANN has been extensively applied in the construction, infrastructure, and manufacturing industry. For example, Zangenehmadar and Moselhi [7] used ANN models to predict the remaining useful life of water pipes in Montreal. Critical factors such as pipe length, diameter, material, and breakage rate were considered to develop robust predictive models. Likewise, Golnaraghi et al. [8] applied the ANN technique, and on that basis, the labor productivity for performing formwork installation operations was predicted. Moon et al. [9] successfully implemented a multilayer perceptron artificial neural network to predict production and latency days for manufacturing production facilities. Although, some studies have advanced the development of forecasting process times at workstations along the production line in modular construction [10]. However, manual and GridSearchCV were used to find the optimal parameters for the machine learning algorithms. This approach is time-consuming and does not ensure optimal parameters. Therefore, various researchers have used GA for hyperparameter tuning for machine learning models in construction and infrastructure engineering. For instance, Assad and Bouferguene [11] used the GA algorithm in order to optimize the hyperparameter of various data mining techniques to accurately predict the water mains condition. Considering these characteristics of GA, the research presented in this paper seeks to implement GA to select the optimal neural network model configurations for predicting the process times of module components in the production line.

2.2 Planning and Scheduling in Modular Construction Manufacturing

Scheduling in the manufacturing industry is a decision-making process where the sequencing of jobs (i.e., module components) and allocation of resources is performed to achieve specific objectives (e.g., minimization of makespan). According to Piendo [12], the scheduling process can be stochastic or deterministic. Since the process times and release dates of the jobs in the production lines are not known with certainty in advance. The scheduling problem in MCM resembles the stochastic scheduling problem. Therefore, various researchers have implemented methods on lean manufacturing principles and simulation to enhance planning and scheduling in MCM. For example, Yu et al. [13] applied a lean-based approach in the modular construction company's production line, improving labor efficiency by 10 % and reducing labor costs by 18%. On the other hand, Moghadam et al. [14] generated multiple scenarios of production line schedules using discrete event simulation. Later, these developed simulation scenarios were integrated with a visualization model to determine the best scenario for balancing the production line. In general, lean and simulation have been applied individually or integrated to develop plans and schedules; thereby improving the performance of MCM. However, there are the following issues: *(i)* a limited number of scenarios were tested; therefore, not giving optimal sequences of modules to be prefabricated in the production line; *(ii)* does not account for decision variables such as module design specifications in their module sequencing arrangement.

2.3 Optimization for Production Scheduling

Various researchers in industrial engineering have applied optimization techniques to solve scheduling problems (e.g., flow and job shop scheduling). For example, Chen et al. [15] proposed a GA-based method that developed schedules for a hybrid flow shop considering order arrivals in a dynamic manner. GA demonstrated its effectiveness by reducing job waiting time and meeting order deadlines. Meanwhile, An et al. [16] illustrated the application of GA in minimizing the production time and cost for a metal-cutting production process. Similarly, optimized schedules were developed for modular and offsite construction. For instance, Hyun et al. [17] developed a multi-objective optimization model based on GA to reduce the duration and cost of modular construction production lines. Rashid and Louis [18] integrated GA and discrete event simulation in order to minimize the total makespan (i.e., completion time) for the modular construction production line by allocating an optimal number of workers at each workstation. However, these developed methods assumed the durations of workstations on a production line as only triangular distribution or determined duration based on number of workers allocated to the activities. There is a lack of scheduling methods that utilize the historical production data while developing predictive models to be used as input for scheduling the sequences of module components in the production line. Although the effectiveness of optimization algorithms depends on the type of scheduling problem and objective, the positive reviews of the GA algorithm (i.e., provides an effective solution and good computing capabilities) lead to its selection to solve MCMSP in this research.

3. Developed Method

Figure 1 presents the proposed method that combines a deep neural network and genetic algorithm in order to solve the modular construction manufacturing scheduling problem (MCMSP). A modular fabricator company collected the historical time data using RFID-based technology [19], consisting of an RFID printer, reader, antenna, and paper-based tags. A detailed discussion of the RFID system design and its application for data was reported in the paper of Sadiq et al. [3]. This research starts with extracting the process times and relevant attributes of module components (e.g., number of studs and doors) at each workstation from the RFID raw data file provided by the company. From the timestamps collected using the RFID system, the process times (i.e., the time required to complete one module competent at each workstation) is extracted based on Equation (1):

$$\text{Processing Time i, w} = \text{First Read Time i, w+1 - First Read Time i, w} \qquad (1)$$

where i = panel; w = workstation; w+1 = next workstation; last read and first read are from antenna description.

In this respect, the next critical step was to perform data preprocessing to clean the data and ensure that the dataset was in the required format for prediction purposes. The first step of data preprocessing is to identify and remove any missing values. In addition, *(i)* data visualization techniques (i.e., pie charts) are applied in order to gain insights into the data, and *(ii)* data points that are above and below 'Mean ± 1.5 SD' are marked as possible statistical outliers. Another crucial preparation step is combining attributes with the same meaning (e.g., the number of single and large doors). Furthermore, normalization techniques are implemented using Equation (2) in order to reduce the sizes of independent variables and model computation time.

$$V' = (V - min_A) \;/\; (max_A - min_A) \qquad (2)$$

where minA is the minimum value and maxA is the maximum value of the independent variable, A, and V represents the original value of A.

The dataset, having been preprocessed, was divided into training (80%) and testing (20%) subsets. Based on the training subset, a deep neural network model, is developed. It should be noted that this study used a deep neural network due to its feature of having multiple processing layers, which can efficiently perform complex nonlinear transformations. Each layer entails several nodes representing the input, transfer, and output variables. In this paper, the rectifier activation function is selected, and the range searched for upper bound/lower bound is 3-10 for hidden layers and 6-100 for the number of nodes. The next critical step is to apply the cross-validation technique in order to prevent overfitting and obtain a better evaluation of the predictive model. The present study adopts K-fold cross-validation for testing the performance of a model. The dataset is divided into K groups, where, in turn, the predictive model is trained using (K−1) groups, and the remaining fold is used to test the accuracy of the model. It should be noted that the parameters of the model (i.e., the number of hidden layers and nodes) have a significant impact on the model's performance. For instance, if there are a small number of nodes, the model cannot be trained well, and with a large number of nodes, performance can be enhanced, but a large number of connections will increase the computational time. Therefore, it is critical to establish parameters for neural networks which can be trained in reasonable computation time and provide errors within the tolerance limit. Therefore, hyperparameter tuning is performed, which is a process of identifying an optimum model configuration. This paper uses the GA optimization technique for hyperparameter tuning to minimize the model's prediction error (i.e., mean absolute error (MAE)). The procedure begins with initializing the population, representing a set of random solutions. The fitness of each solution is calculated, and the best solutions are then selected in the population using a tournament selection strategy. The selected solution is used to reproduce by undergoing the process of crossover and mutations. The process is repeated until optimization criteria are met. It should be noted that only MAE is selected as a measure of goodness of fit, which according to various studies is a better alternative to R square while evaluating the performance of model in respect to nonlinear data [20] and [21]. According to

Lseth [20], utilization of R square for evaluating model performance in nonlinear data, leads to misinterpretations and produces misleading conclusions. Moreover, in the mathematical literature, it has been concluded that the R-square generally do not increase even for better nonlinear predictive models [21]. Later, the unseen data fit into the developed predictive model, and process times were predicted using the deployment function in the python library.

The predictive process times of module components and the type and number of module components were used as input in the optimization model. As mentioned earlier, in a modular construction production line, the duration of each module component at each workstation changes according to its: *(i)* type and design specification (i.e., exterior/interior and number studs, door, and window) and *(ii)* procedure required to prefabricate (i.e., framing, nailing and cutting). At a given production line, the allocated module components are conducted in a cyclic manner (i.e., pre-defined sequence), and the process times of each workstation are a major variable. Therefore, the optimization problem can be defined as a combinatorial problem of the number of module components and process times. To solve the MCMSP, the following assumptions are considered: *(i)* Only module components affect the cycle time of the production line; and *(ii)* module components will be available in the inventory. The problem is modeled as an operation sequencing optimization problem, where $J = \{J_1, J_2 \ldots\ldots J_m\}$ is a set of 'm' number of module module components to be schedules and; $P = \{P_1, P_2 \ldots\ldots P_w\}$ is a set of 'w' number of workstations. Every module component in a modular construction production line needs to be prefabricated on several workstations in the pre-defined order (i.e., sequences), which is considered a non-deterministic polynomial-time hard (NP-hard) problem. That means there are multiple sequences to schedule the module components at workstations, and with an increase in the number of combinations, the complexity of the problem (search space) increases, which makes it harder for the techniques to efficiently find the optimal sequences in a reasonable model runtime. The objective of the optimization problem is to find a near-optimal schedule (i.e., sequences of module components) in order to minimize the makespan (i.e., minimum time to complete all module components in the production line from start to end) [25] represented by equation 3:

$$\text{Min} \left[\text{Max} \left(C_{1,w}, C_{2,w, \ldots\ldots} C_{m,w}\right)\right] \quad (3)$$

Constraints:

$$S_m + D_i \leq S_{i+1} \quad (4)$$

$$\text{Max}\, X_{m,w} = 1 \quad (5)$$

The decision variables are the different sequences of module components in the production line. This optimization problem is subject to a set of constraints: *(i)* constraint 1 (equation 4) defines the production flow of module components, which ensures the module component 'm+1' at station w cannot start before the end of its predecessor module component 'm' at station 'w'; and *(ii)* constraint 2 (equation 5) represents the workstation capacity, which enforces each workstation 'w' can process max of one module 'm' at a time.

The primary component to develop the optimization model is GA, which optimizes the sequences of module components (i.e., order of module components) to be prefabricated at the workstations, which minimizes the makespan of the modular units. The critical elements of the GA algorithm are initialization, evaluation, selection, crossover, and mutation, which works as follows: *(i)* an initial population (i.e., comprised of chromosomes representing a list of module components) is generated randomly. In this research, we have used permutation encoding of modules (i.e., arranging the number of module components in different orders) as a chromosome where a primary module component will be assigned first, then the succeeding module component. For instance, a permutation [4,1,8,5,2,7,3,6] is a chromosome that represents a sequence where module component number 4 gets prefabricated first on all the workstations followed by module component 1, 8, and so on; *(ii)* performing the process of evaluation, where fitness values of each chromosome are evaluated based on the optimization criteria (i.e., minimum makespan); *(iii)* The best chromosomes from the population is selected as the best solution (i.e., optimal sequences of module components) based on their fitness value. In this research, the selection process is performed using roulette wheel selection. It should be noted that the goal of the selection process is to save better and remove bad chromosomes; *(iv)* crossover and mutation are implemented in order to generate new chromosomes for the next generation. For each pair of parents to be mated, a crossover point is chosen using a two-point crossover, and the offspring exchanges the parents' genes among themselves until the crossover point. In the mutation process, to introduce variability and diversity, some of the genes in the offspring are flipped. This selection, crossover, and mutation process continues until the optimization criteria are met.

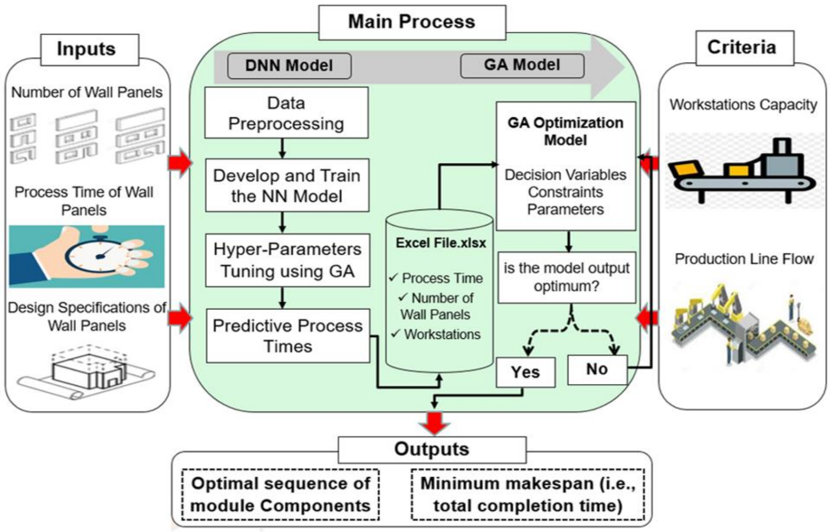
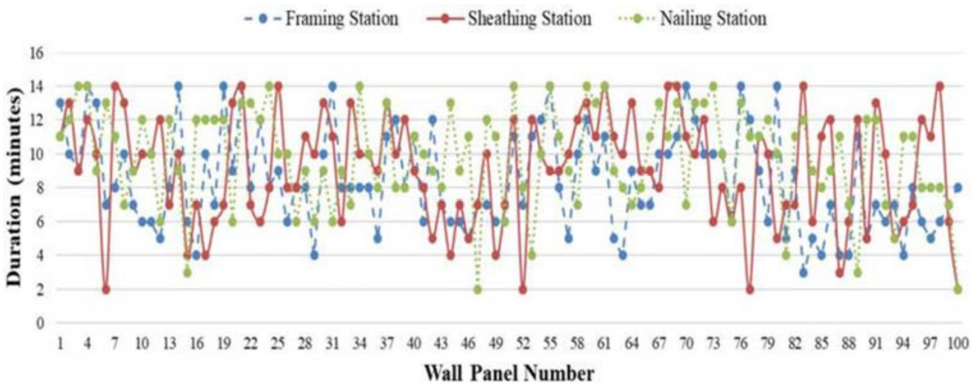


Figure1. Research methodology

4. Case Study and Results

The proposed method was implemented on a wood-frame wall panel production line operated by a modular fabricator in Edmonton, Canada. The wall panel production line consists of the following workstations: *(i)* framing station where the wall components such as studs, tracks, and headers are fastened together to form a wall panel frame; *(ii)* sheathing station where drywall/sheets are installed on wall panels; *(iii)* multifunction bridge where drywalls/sheets are nailed (fixed) and moved to next workstation using transfer cart. It should be noted that interior multiwall panels are moved from the multifunction bridge and are cut at the transfer cart into single-wall panels, which are sent to the window bypass line to store them at the wall magazine line, whereas exterior wall panels are cut into single-wall panels and transferred to the window/door line or window bypass line; *(iv)* Window/Door installation lines, where windows/doors are installed on the wall panels and transferred to the storage area (i.e., wall magazine line). The wall panels are stored at the storage area as they await delivery to sites. Process times at workstations were collected by the modular fabricator company using an RFID system. The data in the 'RFID raw data' file for the workstations include the time between July 2015 and May 2017. As such, it contains *(i)* timestamps for each wall panel along with the workstations; and *(ii)* design factors of each wall panel (e.g., number of studs, number of windows, and length of wall panel). Considering this, the next critical step in the case study was to extract the process times of wall panels at the workstation using Equation (1). In addition, initial data analysis was performed in order to gain insights into the production line. Figure 2a shows a high level of variance in the process times at workstations due to the influence of the design factors (i.e., number of windows, panel length, and number of studs). For example, at the sheathing station, the processing time of wall panel 3 was 9 minutes, wall panel 4 was 12 minutes, and wall panel 6 was 2 minutes. This variation in process times affects the daily productivity of the workstations. Moreover, Figure 2b illustrates that the daily production on March 30 was 12 panels at the sheathing station, 22 panels at the framing station, and 6 panels at the nailing station, respectively, causing an imbalanced production line.



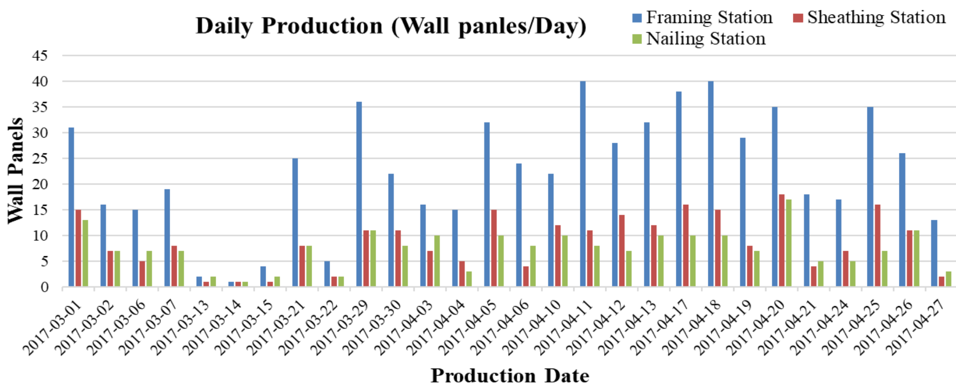


Figure 2 (a): Process times of wall panels at workstations; and (b): Daily production of wall panels

Prior to the development of a predictive model, data preprocessing was implemented. Timestamps of wall panels prefabrication that started on one day and finished on the following day were removed, and wall panels with missing timestamps were also discarded from the dataset. Additionally, similar properties of a panel were combined into a single attribute in order to reduce data dimensions. For example, DStud, LStud, and MStud were combined into a 'stud'; similarly, window and large widow were combined as a 'window.'

Later the outliers were removed based on data visualization (i.e., pie chart), which helps to visualize the distribution of data points that are inconsistent from the data set. For example, at butterfly (i.e., cutting workstation), the process times above 60 minutes were removed (Figure 3). The reason for removing these points is that around 4% of the wall panel's processing times have excessive times (i.e., 61-410000 minutes). Such data points indicated a work disruption due to errors in the shop drawings and resulted from their waiting between the workstations. In addition, data points above and below 'Mean ± 1.5 SD' are data points marked as possible statistical outliers. As a result of the preprocessing tasks, the datasets numbered 7256, 2885, 3035, 19998, 1868, 4101, and 1592 for the framing, sheathing, nailing, cutting, window door, window bypass, transfer table, and storage area workstations, respectively.

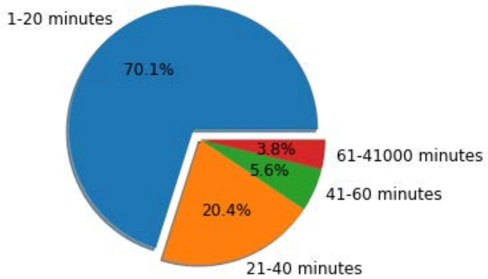


Figure 3. Pie chart

Next, the min-max normalization technique was applied in order to transform values ranging between 0 and 1. As described above, the variation in the process times at workstations due to differences in design factors of wall panels leads to an imbalanced production line. In this respect, the next critical step in the case study was to develop a predictive model by considering the wall panel design factors.

The developed deep neural network consists of input, hidden, and output layers. The independent variables, such as panel length, number of regular studs, number of doors, and number of windows, were used as model inputs, while the process times of wall panels in minutes were the output variable. Each node element was connected and layered with nodes of the next layer. The nodes, which carry weights, were used to process the error rate. In this paper, the rectifier activation function was selected, and the range searched for upper bound/lower bound was 3-10 for hidden layers and 6-100 for the number of nodes.

In order to identify the optimal number of hidden layers and nodes, a GA optimization algorithm was selected to minimize the MAE. In this paper, the optimization parameters were assigned as follows: *(i)* population size of 20; *(ii)* the maximum number of generations was 50; *(iii)* mutation probability of 0.1; *(iv)* crossover probability of 1 and: *(v)* number of tournaments were 3. As observed in table 1, most of the workstations (i.e., framing, sheathing, and nailing station) had MAE of less than 2.50 minutes, respectively.

Table 1. Selected values of the predictive model

Workstations	**Selected Value**		**MAE**
	Hidden	**Nodes**	
Framing	3	74	2.17 min
Sheathing	3	72	2.11 min
Nailing	3	14	2.41 min
Butterfly (cutting)	7	70	5.37 min
Window/Door	8	42	30.36 min
Window Bypass	3	86	20.16 min
Transfer Table	9	24	1.11 min

Later the predictive process times were used as input in the optimization model. A genetic algorithm was developed with an objective of minimizing the makespan (i.e., production duration) of prefabricating the wall panels in the production line. The number of wall panels considered for this optimization model are 46 (i.e., 27 exteriors and 19 interiors) to be processed at 7 workstations, respectively. The optimization parameters used in this case study are as follows: 100 generations, each generation containing 30 populations with mutation and crossover rates of 0.2 and 0.8. Figure 4 shows that the Makespan value (i.e., duration) plateaued after 85 generations. The algorithm starts with an initial solution of a makespan equal to 4359 minutes, and the algorithm converges to the best solution in 90 generations with a makespan equal to 4352 minutes.

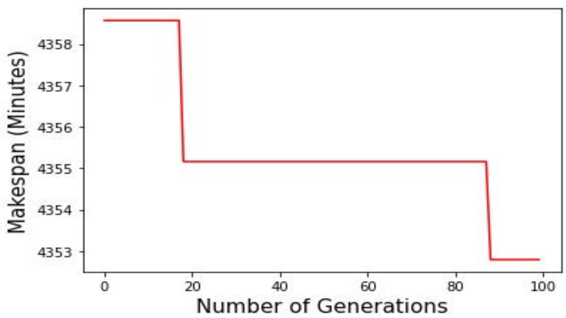


Figure 4. Makespan value

The median makespan to prefabricate 46 wall panels is 4352 min (i.e., 72.32 hours ~ 7 days (10 hours/day working time for the factory). The optimal sequence is: 9,16,25,14,23,10,44,42,27,5,45,36,26,28,11,13,31,22,29 ,18,39,38,43,6,22,40,30,2,4,17,12,33,8,34,15,32,41,7,1, 4,2,21,3,19,35,37, where the wall panel number 9 is the first panel to be prefabricated at the first station (i.e., framing) followed by 16 and so on. The approach presented in this paper unleashes the potential of a data driven scheduling method, which overcomes the drawbacks of previous studies with respect to: *(i)* testing limited number of scenarios for sequencing module components, thus not providing near optimal solution and; *(ii)* assuming the durations on a production line as only triangular distribution or based on number of workers allocated to the activities.

5. Conclusions and Future Work

In modular construction, the module components (e.g., wall, roof, and floor panels) are of various sizes and design specifications, necessitating dynamic changes to the production line. This poses a challenge for production line managers to accurately forecast the process times of module components at each workstation, leading to inefficient production line schedules and reduced productivity. In this respect, this research proposes a newly developed method that utilizes deep neural network and GA in order to solve the MCMSP. The developed method utilizes the historical time data collected from the manufacturing plant at each workstation as an input in the deep neural network model to predict the process times at these workstations. Next, the GA optimization technique is implemented for hyperparameter tuning and, as such, finds the optimal number of hidden layers and a number of nodes in each layer. Subsequently, the predicted process times are used as an input in the scheduling optimization model in order to provide optimal sequence of module components to minimize the makespan. The case study implementation of the developed method in a wall panel production line demonstrates that the developed method predicted the process times with a MAE of less than 2.50 minutes for most of the workstations. In particular, the optimal sequences of 46 wall panels are prefabricated in 72.32 hours. As illustrated by the case study, this method can be helpful in assisting production managers to understand the insights of the production line by: *(i)* predicting process times for various types of module components; and *(ii)* analysing variance in the process times at workstations due to the influence of the module components design factors. In this way, production line managers can reduce duration of prefabricating module components at workstations and don't need to rely on experienced based approach with respect to estimating the process times and developing schedules.

However, the developed method does have limitations in the following respect: *(i)* other factors important to develop schedules for production line besides duration, (e.g., minimization of the idle time at workstations and allocation of optimal number of workers at workstations), are not taken into consideration in the current method; and *(ii)* the number of module components considered for this optimization model are not based on multiple projects. Therefore, in order to improve the current method, the future work will seek to expand in the following directions: *(i)* optimization model will consider module components (e.g., wall panels) based on multi-projects to achieve JIT lean strategy; *(ii)* the optimization model can be further improved by applying hybrid optimization techniques; and *(iii)* multi-objective optimization of production line schedules with respect to cost and workers will be explored. It should be noted that this paper focused primarily on developing production line schedule with an objective of minimizing duration, which is an essential factor while solving sequencing problems for production line. However, consideration of quality of work can also be incorporated into future works by measuring the percentage of rework for prefabricating module components at workstations.

Acknowledgments

The authors wish to acknowledge the support of Dr. Mohammed Sadiq Altaf in providing the production line data. Special thanks to Hassan Bardareh for helping in automating RFID-based data extraction of the production line.

References

[1] Nik-Bakht M., An C., Ouf M., Hafeez G., Dziedzic R., Han S.H., Nasiri F., Eicker U., Hammad A. and Moselhi O. Value Stream Mapping of Project Lifecycle Data for Circular Construction. In *Proceedings of the International Symposium on Automation and Robotics in Construction*, 38, 1033-1042, Dubai, United Arab Emirates, 2021.

[2] Salama T., Moselhi O. and Al-Hussein M. Overview of the Characteristics of the Modular Industry and Barriers to its Increased Market Share. *International Journal of Industrialized Construction*, 2(1), 30-53, 2021.

[3] Altaf M.S., Bouferguene A., Liu H., Al-Hussein M. and Yu H. Integrated production planning and control system for a panelized home prefabrication facility using simulation and RFID. *Automation in construction*, 85, 369-383, 2018.

[4] Chu W., Han S.H., Zhen L., Hermann U. and Hu D. A Predictive Model for Scaffolding Man-hours in Heavy Industrial Construction Projects. In *Proceedings of the International Symposium on Automation and Robotics in Construction*, 37, 976-983, Kitakyushu, Japan 2020.

[5] Hegazy T., Fazio P. and Moselhi O. Developing practical neural network applications using backpropagation. *Computer Aided Civil and Infrastructure Engineering*, 9(2), 145-159, 1994.

[6] Moselhi O., Hegazy T. and Fazio P. Neural networks as tools in construction. *Journal of construction engineering and management*, 117(4), 606-625,1991.

[7] Zangenehmadar Z. and Moselhi O. Assessment of remaining useful life of pipelines using different artificial neural networks models. *Journal of performance of constructed facilities*, 30(5), 04016032, 2016.

[8] Golnaraghi S., Zangenehmadar Z., Moselhi O. and Alkass S. Application of artificial neural network (s) in predicting formwork labour productivity. *Advances in Civil Engineering*, 2019.

[9] Moon S., Hou L. and Han S. Empirical study of an artificial neural network for a manufacturing production operation. *Operations Management Research*,1-13, 2022.

[10] Mohsen O., Mohamed Y. and Al-Hussein M. A machine learning approach to predict production time using real-time RFID data in industrialized building construction. *Advanced Engineering Informatics*, 52, 101631, 2022.

[11] Assad A. and Bouferguene A. Data Mining Algorithms for Water Main Condition Prediction Comparative Analysis. *Journal of Water Resources,* 148(2), 04021101, 2022

[12] Pinedo M.L., 2012. Scheduling, (Vol. 29). New York: Springer.

[13] Yu H., Al-Hussein M., Al-Jibouri S. and Telyas A. Lean transformation in a modular building company: A case for implementation. *Journal of management in engineering*, 29(1),103-111, 2013.

[14] Moghadam M., Barkokebas B. and Al-Hussein M. Post-simulation visualization application for production improvement of modular construction manufacturing. In *Proceedings of the International Symposium on Automation and Robotics in Construction*, 31,1, Sydney, Australia, 2014

[15] Chen J., Wang M., Kong X.T., Huang G.Q., Dai Q. and Shi G. Manufacturing synchronization in a hybrid flowshop with dynamic order arrivals. *Journal of Intelligent Manufacturing*, 30(7), 2659-2668, 2019.

[16] An L., Yang P., Zhang H. and Chen M. Multi-objective optimization for milling operations using genetic algorithms under various constraints. *International Journal of Networked and Distributed Computing*, 2(2), 108-114, 2014.

[17] Hyun H., Yoon I., Lee H.S., Park M. and Lee J. Multi-objective optimization for modular unit production lines focusing on crew allocation and production performance. *Automation in Construction*, 125, 103581, 2021.

[18] Rashid K., Louis J. and Swanson C. Optimizing labor allocation in modular construction factory using discrete event simulation and genetic algorithm. *In Proceedings of Winter Simulation Conference,* 2569-2576, 2020.

[19] Bardareh, H. and Moselhi, O. An integrated RFID–UWB method for indoor localization of materials in construction. *Journal of Information Technology in Construction*, 27(32), 642-661, 2022

[20] Kva Lseth, T.O. Note on the R 2 measure of goodness of fit for nonlinear models. *Bulletin of the Psychonomic Society*, 21, 79-80, 1983.

[21] Spiess, A.N. and Neumeyer, N. An evaluation of R2 as an inadequate measure for nonlinear models in pharmacological and biochemical research. *BMC pharmacology*, *10*(1), 1-11, 2010.