# ARRANGING PRECAST PRODUCTION SCHEDULES USING GENETIC ALGORITHMS

**Chien-Ho Ko†**
Department of Civil Engineering,
National Pingtung University of Science and Technology,
Pingtung, 91201, TAIWAN
+886-8-770-3202, Email: ko@mail.npust.edu.tw

**Shu-Fan Wang**
Department of Industrial Engineering and Technology Management,
Da-Yeh University, Changhua 51591, TAIWAN

## Abstract

The goal of production scheduling is to achieve a profitable balance among on-time delivery, short customer lead time, and maximum utilization of resources. However, current practices in precast production scheduling are fairly basic, depending heavily on experience, thereby resulting in inefficient resource utilization and late delivery. Certain computational techniques have been proven effective in scheduling. To enhance precast production scheduling, this research develops a Multi-Objective Precast Production Scheduling Model (MOPPSM). In the model, production resources and buffer size between stations are considered. A multi-objective genetic algorithm is then developed to search for optimum solutions with minimum makespan and tardiness penalties. The performance of the proposed model is validated by using five case studies. The experimental results show that the MOPPSM can successfully search for optimum precast production schedules. Furthermore, considering buffer sizes between stations is crucial for acquiring reasonable and feasible precast production schedules.

**Keywords:** Precast production, Scheduling, Multi-objective genetic algorithms, Buffer

## 1. INTRODUCTION

Precast construction is a method to build up constructions by prefabricated concrete elements (Demiralp et al. 2012). Precast fabricators deliver elements to construction site according to its erection schedule. Building up constructions using precast elements can reduce uncertainty than those casted in the construction site. In addition, it fits in with the needs of industrial process. As a result, precast fabrication in the construction industry can be categorized as manufacturing (Hossain and Ozyildirim 2013). Production scheduling is one of the most important tasks in the manufacturing. Different production schedule can induce different throughput. Industrial engineers therefore endeavor to finish products with a minimum makespan (Tharmmaphornphilas and Sareinpithak 2013). To enhance competitiveness, production schedulers face the challenge to satisfy multiple objectives since one objective may conflict with the others.

The current practice of making precast production scheduling depends on scheduler's experience. However, manually arranging production schedules frequently results late

† Corresponding author

delivery and wastes production resources (Dawood 1993, Chan and Hu 2002). Recently, researchers have started on using computational techniques to deal with scheduling issues. Chan and Hu (2002) developed a production model based on flowshop production. A Genetic Algorithm (GA) was used to solve the model. In their research, production activities were categorized into interruptible and uninterrupted groups. Benjaoran et al. (2005) proposed a flowshop sequencing model with a multi-objective GA. Multiple objectives in their study include minimum machine idle time, minimum late delivery penalty, and minimum makespan. Previous studies have proven that precast production is a flowshop production. In addition, production resources have a crucial impact on throughput.

## 2. PRECAST PRODUCTION PROCESS

Precast production is a flowshop production that can be divided into 6 steps, namely mold assembly, placement of reinforcement and all embedded parts, concrete casting, curing, mold stripping, and product finishing. The process is depicted in Figure 1. Mold assembly is to provide mold with a specific dimension for element. In general, fabricators use steel mold for a durable purpose. Precast concrete primarily contains two kinds of materials i.e. concrete and steel bars. Reinforcement and embedded parts are placed in their positions after the mold is complete. Embedded parts are used to connect and fix with other elements or with structure when precast elements are assembled. Concrete is cast when everything inside the element is in the right place. To enhance the chemistry solidifying concrete, curing concrete with steam is implemented. Otherwise, concrete takes weeks to reach its legal strength. Moving, erecting, or assembling elements before legal strength may damage elements. Molds can be striped after the concrete becomes solid. Due to the cost of developing steel molds, fabricators reuse them once they are stripped. The final step of production is product finishing. Minor defects such as scratch, peel-offs, uneven surfaces, etc. are treated in this step.

Traditional flowshop sequencing problem regarded production as a continuous flow. However, precast production owns activities that can be done after working hours. Typical equation shown in Equation (1) that used to calculate completion time cannot meet the needs in the precast industry.

$$C\left(J_j, M_k\right) = Max\left\{C(J_{j-1}, M_k), C(J_j, M_{k-1})\right\} + P_{jk} \tag{1}$$

Notations used in Equation (1) are explained as follows:

$C\left(J_j, M_k\right)$: Completion time for jth element in k machine.

$P_{jk}$ : Operation time for jth element in k machine, $P_{jk} \geq 0$.

Equation (1) assumes an infinite buffer size between stations. Due to the large size of precast elements, Equation (1) is reformulated as Equation (2).

$$C\left(J_j, M_k\right) = Max\left\{C(J_{j-1}, M_k) + WT_{j-1,k}, C(J_j, M_{k-1})\right\} + P_{jk} \qquad (2)$$

where $WT_{j-1,k}$ is the time for (j-1)th element in k machine waiting to be sent to buffer.

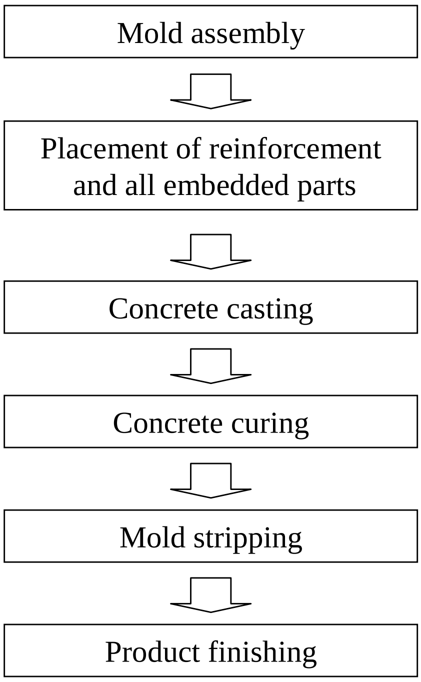


**Figure 1**: Precast production process.

The Gantt chart of precast production is illustrated in Figure. 2. In the production, interruptible activities including mold assembly, placement of parts, mold stripping, and finishing can be done by the next day. Curing is categorized as uninterruptible activity that must be doing continuously until completion. Curing is a special task differing from other manufacturing. It is a time-consuming task and is frequently completed by machines without workers. As a result, it can be arranged in any time, even after the hours of working day. The other special requirement for curing is that it must be done right after casting i.e. no wait.

Molds are necessary for precast fabrication. Number of molds is a crucial constrain for production scheduler. Due to the high cost of steel mold, fabricators only develop a few molds. As a result, makespan and throughput are harnessed by number of molds. For example, due to a limited number of mold A, element 3 with mold A cannot be started fabrication until element 1 releases mold A. The example demonstrates a situation that fabrication waits for mold, which frequently happens in practice. In the process of scheduling, sequence of molds is arranged according to the number of molds and types of molds.

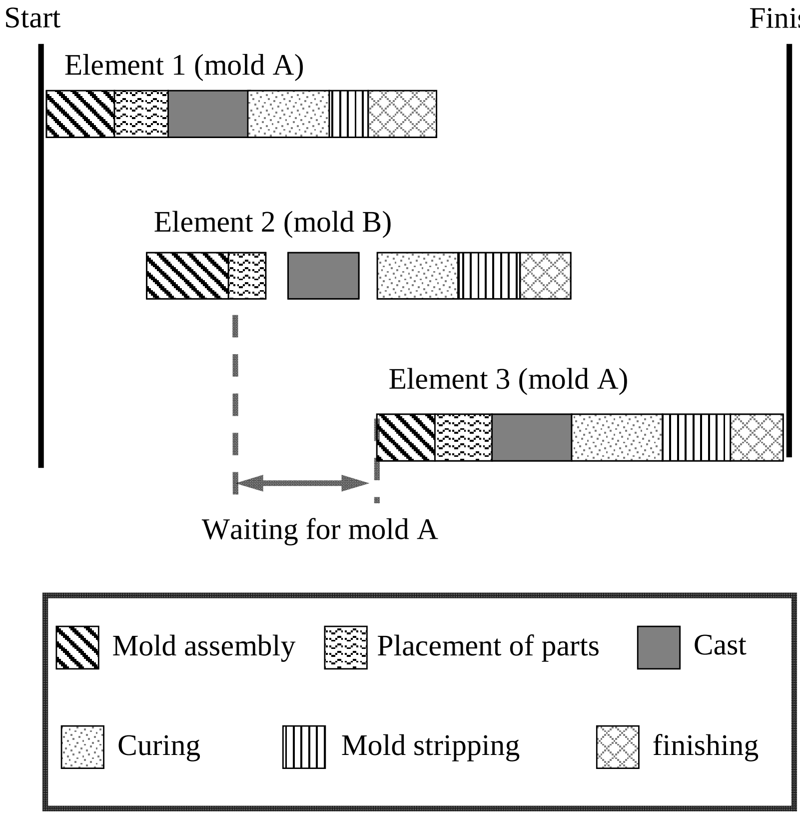


**Figure 2:** Gantt Chart of precast production.

## 3. MULTI-OBJECTIVE GENETIC ALGORITHM

The study adopts the Multi-Objective Genetic Local Search Algorithm (MOGLS) proposed by Ishibuchi and Murata (1998) to search for optimum production schedules. The evolutionary process of MOGLS is represented in Figure 3. Each step is discussed in the following sections.

### Step 1: Encoding

Factors effect precast makespan include production resources and production sequence. Some production resources such as number of cranes and factory size cannot be changed by s chedulers. Others such as buffer size between stations, mold number, and working hours can b e determined by them. The study encodes production schedule by job sequence. Buffer sizes a nd mold amount are treated as production constraints while scheduling.

### Step 2: Initializing Population

The variation of initial solution with higher fitness value can improve searching efficiency. To provide an equal chance for every state space, a set of initial solutions are randomly generated. Those chromosomes offer a base for further evolutionary process.

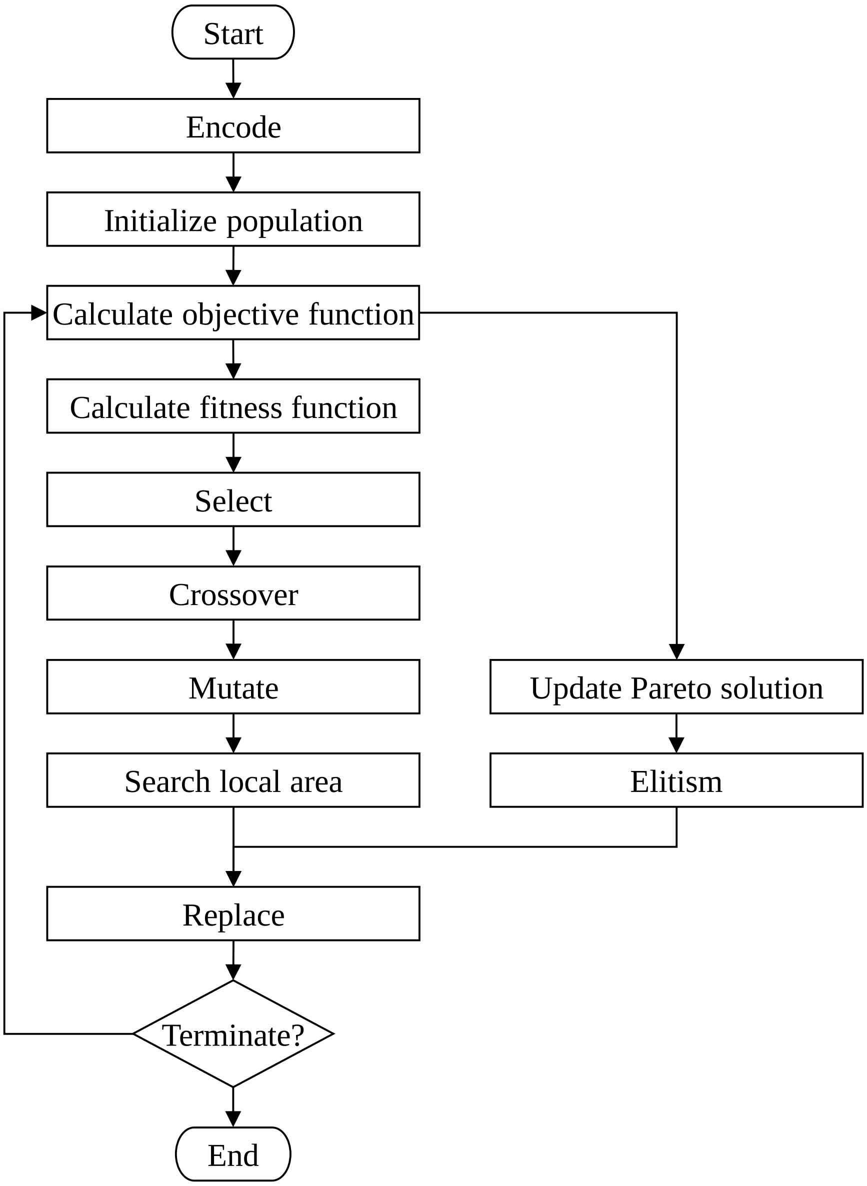


**Figure 3**: Evolutionary process of MOGLS.

## Step 3: Calculating Objective Function

In this step, chromosomes are decoded corresponding with precast production model. Two objectives are considered in the study: minimum makespan and minimum cost of penalty. The objective function is displayed in Equation (3)

$$f(x) = \omega_1\left(f_1(x)\right) + \omega_2\left(f_2(x)\right) \tag{3}$$

where $\omega_1$, $\omega_2$ are positive weights and $\omega_1 + \omega_2 = 1$. $f_1(x)$ is a makespan function and $f_2(x)$ is a penalty function.

## Step 4: Updating Pareto Solution

To make sure that derived solutions conform to the definition of Pareto solution, every generation has to update Pareto solution pool. The way to update the pool is to put the chromosomes that conform to the definition of Pareto solution in the Pareto solution pool.

## Step 5: Calculating Fitness Function

To evaluate the fitness of each chromosome, objective value is converted to fitness value. In multi-objective programming, since distribution of each objective value is deferent, each objective value is normalized in advance. Then, a weighted-sum method can be applied. Cochran et al. (2003) proposed that sub-objectives are normalized by its fittest value. Equation (4) is thus used to convert fitness value.

$$f(x) = \omega_1 \left(\frac{f_1(x)}{f_1^*(x)}\right) + \omega_2 \left(\frac{f_2(x)}{f_2^*(x)}\right) \tag{4}$$

where $f_1^*, f_2^*$ represent the minimum makespan and minimum cost of penalty in the initial solution individually.

## Step 6: Selecting

Selection operator is used to select chromosome according to its fitness. Higher fitness value has higher chance for survival. The purpose of the selection operator is to choose fitter chromosomes for evolving better generations. The study adopts roulette-wheel method for selection (Goldberg 1989). For population size $N_{pop}$ and elitism number $N_{elite}$, every generation selects $\left(N_{pop} - N_{elite}\right)$ chromosomes.

## Step 7: Crossover

GA extends searching space by crossover operator. The operator produces next generation by exchanging partial information of parents. The resulting generation represents a new set of solution. This study uses two-point crossover that randomly determine two points. Genes between the two points remain. The other parts are exchanged.

## Step 8: Mutation

The mutation operator produces spontaneous random changes in various chromosomes. It protects against premature loss of important notations. The study uses shift mutation that randomly selects two points. The rear point inserts in advance of the prior point, then the whole gene shift back forward.

### Step 9: Elitism

Elitism has been proven successful in GA (Goldberg 1989). It survives a certain amount of Pareto solution to the next generation. So every generation contains elite solutions for better evolvement. Applying the strategy, fitness increases generation by generation.

### Step 10: Replacement

Replacement is a process that produced chromosomes eliminate parent chromosomes. In the process, previous population is renewed by generated offspring. Therefore, next generation can continuously involve new solutions for evolvement.

### Step 11: Terminate Conditions

Terminate conditions provide criterion for stopping evolutionary process. In general, evolutionary process is terminated by iterations and/or required fitness. This research terminates the evolutionary process by iterations assigned by users.

## 4. EXPERIMENT

The study experiments the efficiency of applying GA in precast production scheduling. Single objective GA, multi-objective GA, and multi-objective GA with a finite buffer size are experimented.

### 4.1 Single Objective GA

The study firstly experiments the efficiency of applying single objective GA in precast production scheduling. Production data shown in Table 1 are acquired from Chan and Hu (200 2).

MOGLS can be applied to solve single objective problem. GA parameters for the problem are explained as follows:

- Population size: 10
- Termination condition: 1000 iterations
- Crossover rate: 0.9
- Mutation rate: 0.005
- Elite number: 1
- Local search: disable

Since it is a small problem, local search is disabled. Experiment results are shown in Table 2. $GA^1$ in the table denotes MOGLS with single objective “makespan” whereas $GA^2$ represents MOGLS with single objective “cost of penalty.” Observing the table, the schedule for single objective makespan, i.e. 4-5-2-6-1-3, is found by the algorithm. Schedule (i.e. 4-2-3-1-6-5) with minimum cost of penalty 24.6 can be found by the algorithm too. The experiment validated that GA is promising to optimize precast production schedules.

**Table 1**: Production data for single objective GA.

| Element | Production time | | | | | Due day (h) | Penalty | |
|---|---|---|---|---|---|---|---|---|
| | Mold assemble | Parts placement | Concrete casting | Mold stripping | Product finishing | | Inventory | Late delivery |
| 1 | 1 | 0.8 | 1.2 | 1.5 | 0.5 | 28 | 2 | 10 |
| 2 | 1.7 | 2 | 2 | 1.5 | 2.5 | 28 | 2 | 10 |
| 3 | 0.4 | 0.5 | 0.6 | 0.5 | 0 | 28 | 1 | 10 |
| 4 | 0.3 | 0.4 | 0.5 | 0.4 | 1 | 28 | 1 | 10 |
| 5 | 1.5 | 1.8 | 1.2 | 1.5 | 1.5 | 52 | 2 | 10 |
| 6 | 1.5 | 1.6 | 1.5 | 1.8 | 0.8 | 52 | 2 | 10 |

**Table 2**: Experiment results for single objective problem.

| Solver | Production sequence | Makespan (h) | Cost of penalty |
|---|---|---|---|
| GA[1] | 4-5-2-6-1-3 | 48.5 | 310.4 |
| GA[2] | 4-2-3-1-6-5 | 51.0 | 24.6 |

## 4.2 Multi-Objective GA

The effectiveness of single objective GA in precast production scheduling has been proven in the previous section. This section discusses the promising of using MOGLS with multi-objective in precast production scheduling.

Production data experimented in this section are acquired from Benjaoran et al. (2005) (see Table 3). In this case, prefabricator has two A molds, two B molds, and one C mold. The experiment includes ten elements, which provides 10! combinations. To compare experiment results, 26 Pareto solutions have been discovered through the principle of exhaustion. Ten of 26 Pareto solutions are demonstrated in Table 4.

GA parameters implemented in the experiment are summarized below:

- Population size: 20
- Termination condition: 2000 iterations
- Crossover rate: 0.9
- Mutation rate: 0.005
- Elite number: 4
- Local search: 2 times

**Table 3**: Production data for multi-objective GA.

| Element | Production time | | | | | Mold type | Due day (h) | Penalty | |
|---|---|---|---|---|---|---|---|---|---|
| | Mold assemble | Parts placement | Concrete casting | Mold stripping | Product finishing | | | Inventory | Late delivery |
| 1 | 2 | 1.6 | 2.4 | 2.5 | 1 | A | 112 | 2 | 10 |
| 2 | 3.4 | 4 | 4.0 | 2.4 | 5 | B | 112 | 2 | 10 |
| 3 | 0.8 | 1 | 1.2 | 0.8 | 0 | A | 112 | 1 | 10 |
| 4 | 0.6 | 0.8 | 1.0 | 0.6 | 2 | A | 112 | 1 | 10 |
| 5 | 3 | 3.6 | 2.4 | 2.4 | 3 | C | 208 | 2 | 10 |
| 6 | 3 | 3.2 | 3.0 | 3 | 1.6 | A | 128 | 2 | 10 |
| 7 | 1.3 | 0.9 | 2.4 | 1.9 | 1.8 | C | 144 | 2 | 10 |
| 8 | 1.7 | 1.4 | 1.1 | 0.9 | 0.7 | B | 144 | 2 | 20 |
| 9 | 2.2 | 1.8 | 1.2 | 2.3 | 0.7 | A | 144 | 1 | 20 |
| 10 | 1.6 | 3.2 | 2.3 | 2.1 | 2.7 | C | 240 | 1 | 20 |

**Table 4**: Ten example of Pareto solutions.

| No. | Production sequence | Makespan (h) | Cost of penalty |
|---|---|---|---|
| 1 | 5-6-10-4-2-9-7-1-8-3 | 100.2 | 1024.3 |
| 2 | 5-4-2-10-1-6-8-7-9-3 | 99.4 | 1165.2 |
| 3 | 5-4-2-10-9-6-8-7-1-3 | 99.6 | 1142.3 |
| 4 | 5-4-3-2-10-6-1-8-7-9 | 99.3 | 1181.6 |
| 5 | 8-2-4-10-3-1-6-9-7-5 | 173.4 | 594.2 |
| 6 | 8-2-4-1-3-9-6-7-5-10 | 220.8 | 458 |
| 7 | 8-2-4-3-1-6-9-7-5-10 | 196.8 | 503.4 |
| 8 | 9-10-3-2-6-5-4-8-1-7 | 102.2 | 1005.5 |
| 9 | 9-10-4-6-5-1-2-3-7-8 | 121.9 | 858.1 |
| 10 | 9-10-4-3-2-5-1-6-8-7 | 103.6 | 977.7 |

Since GA is a stochastic search, every execution produces different result. To verify the performance of the method, experiment result displayed in Table 5 is an average for 20 runs. Observing the accuracy, MOGLS is good enough for arranging precast production schedules comparing to currently manual practice.

**Table 5**: Experiment results for multi-objective problem.

| Solver | Derived number of Pareto solution | Correct number of Pareto solution | Accuracy |
|---|---|---|---|
| MOGLS | 21.34 | 17.62 | 82.56% |

## 4.3 Multi-Objective GA with A Finite Buffer Size

Unlike regular manufacturing, precast elements occupy large spaces. It is not reasonable if buffer sizes between stations are ignored. Otherwise, production schedules are not realistic since fabricator provides precast fabrication with a finite space. The objective of this section is to validate that buffer sizes between stations is one of crucial constraints for schedulers.

Production data shown in Table 3 are used for experiment. GA parameters are the same with previous section. Maximum buffer sizes between stations are set as five. Experiment results are shown in Table 6.

**Table 6**: Experiment results for multi-objective problem with a finite buffer.

| Buffer size | Makespan | Penalty | Required buffer size |
|---|---|---|---|
| 5 | 126.9 | 701.6 | 2 |
| 4 | 126.9 | 701.6 | 2 |
| 3 | 127.1 | 706.2 | 2 |
| 2 | 132.3 | 717.9 | 2 |
| 1 | 134.7 | 729.1 | 1 |

Observing the results, maximum required buffer size for the production system is two. Therefore, buffer size has no impact on makespan and cost of penalty when buffer size is more than two. By the contrast, if buffer size is smaller than the required buffer size, both makespan and cost of penalty increases.

# 5. CONCLUSIONS

The study describes precast production process with a mathematical model. A multi-obje ctive GA developed based on MOGLS is proposed to solve the model. Multi-objective consid ered in the study is to minimize makespan as well as cost of penalty. Three experiments are us ed to demonstrate the performance of GA in single objective, multi-objective, and multi-objec tive with a finite buffer size at precast production scheduling. Implementation results show th at multi-objective GA can offer production schedulers with a set of subjective Pareto solutions . The information provided by GA can assist schedulers to make proper production schedules.

Numbers of molds is one of important factors directly impacting production schedules. Therefore, how to determine optimum numbers of molds with production scheduling should be studied in the future.

## REFERENCES

Benjaoran, V., Dawood, N., and Hobbs, B. (2005) Flowshop scheduling model for bespokeprecast concrete production planning. *Journal of Construction Management and Economics*, 23 (1), 93-105.

Chan, W. T. and Hu, H. (2002) Constraint programming approach to precast production scheduling. *Journal of Construction Engineering and Management, ASCE*, 2002, 128 (6), 513-521.

Chan, W. T. and Hu, H. (2002) Production scheduling for precast plants using a flow shop sequencing model. *Journal of Computing in Civil Engineering, ASCE*, 79 (17), 1605-1616.

Cochran, J. K., Horng, S. M., and Fowler, J. W. (2003) A multi-population genetic algorithm to solve multi-objective scheduling problems for parallel machines. *Computers and Operations Research,* 30 (7), 1087-102.

Dawood, N. N. (1993) Knowledge elicitation and dynamic scheduling using a simulation model. An application to the precast manufacturing process. *Proceedings of the Civil-Comp93, Part 4: Knowledge Based Systems for Civil and Structural Engineering*, 73.

Demiralp, G., Guven1, G., Ergen, E. (2012) Analyzing the benefits of RFID technology for cost sharing in construction supply chains: A case study on prefabricated precast components, *Automation in Construction*, 24, 20–129.

Goldberg, D. E. (1989) *Genetic Algorithms in Search, Optimization, and Machine Learning*, Addison-Wesley, Reading, Mass.

Hossain, M. S. and Ozyildirim, H. C. (2013) Use of Precast Slabs in Rapid Construction, *Concrete International*, 35 (2), 41-46.

Ishibuchi, H. and Murata, H. (1998) Multi-objective genetic local search algorithm and its applications to flowshop scheduling. *IEEE Transactions on SMC*, 28, 392-403.

Tharmmaphornphilas, W. and Sareinpithak, N. (2013) Formula selection and scheduling for precast concrete production, *International Journal of Production Research*. DOI: 10.1080/00207543.2013.795250