

Crane Scheduling in Steelmaking Plants via Priority Propagation and a Memetic Algorithm

Yongnian Mao¹, Qiuhua Tang^{1*}, Liping Zhang¹ and Qimin Zhang^{1,2}

¹*Industrial Engineering Department, Wuhan University of Science and Technology, Wuhan 430081, China*

²*General plant of bar stock, Wuhan Iron and Steel (Group) Company, Wuhan 430082, China*

tangqiuhua@wust.edu.cn

Abstract

In steelmaking and continuous casting plants, several cranes installed on a common track are expected to perform multiple conveyance tasks effectively and stably. This paper states a multiple-crane scheduling problem with three objectives of maximizing the number of the completed tasks, minimizing the total transport distance, and minimizing the total passive transport distance. In light of the limitation of the workshop layout, the material properties and the production processes, each task is divided into five categories and prioritized, and then a priority propagation procedure is designed to avoid the potential collisions among cranes by forcing one crane making way for another whenever interference exists. This priority propagation based collision avoidance mechanism is embedded into the decoding process of a memetic algorithm to ensure the feasibility of solutions. And the algorithmic operators including crossover and local searches are calibrated to enforce an efficient tradeoff of intensification and diversification. The computational experiments demonstrate that the proposed priority propagation and memetic algorithm are capable of ensuring the smoothness of cranes and improving the efficiency of production.

Keywords: Crane Scheduling, Priority Propagation, Memetic Algorithm, Local Search

1. Introduction

Crane is material-handling device employed in factories to lift or lower heavy materials and to move them horizontally. In general, several cranes are placed on a common track, causing potential collisions among cranes during operations. The scheduling of multiple cranes aims to deliver the materials on time, ensure the continuity of production and smooth the movement of cranes.

In most steelmaking plants, the crane scheduling is performed manually by specific personnel called dispatchers. Their job duty necessitates years or decades of professional experience in production planning and scheduling. However, when several tasks are expected to be performed, the crane scheduling problem is almost intractable even for senior dispatchers. Moreover, cranes have to change directions frequently so as to avoid potential collisions among neighboring cranes. All in all, manual crane scheduling cannot provide precise information for the operations of cranes, and it is important to perform research on the methodology of the crane scheduling.

For crane scheduling problem, the cranes can be divided into several sub-sets due to the special circumstances: quay cranes [1-5], yard cranes [6-10], hoist cranes [11-13] and cranes in steel-making continuous-casting process [14] and so on. For quay cranes, Kim and Park [1] formulated a mixed integer linear programming model. Moccia *et al.* [2] revised the model proposed by Kim and Park (2004) and proposed a branch and cut

algorithm. Sammarra *et al.* [3] proposed a tabu search method. Bierwirth and Mesisel [4] presented a heuristic called unidirectional scheduling based on a tree search. Chung and Choy [5] proposed a modified genetic algorithm. For the quay cranes, Kim and Kim [6] proposed a mixed integer programming model (MIP) and Kim and Kim [7] employed heuristic algorithms. Additional contributions to MIP and heuristic algorithms for yard cranes deployment can be found in [8-10]. In regard to hoist cranes in electroplating production line, a “zone partition” method was used by [11-13] to transform the multi-hoist scheduling problem into single-hoist scheduling problem. For the crane scheduling in steelmaking process, Tanizaki *et al.* [14] presented a mathematical model and a backward-forward simulation was used to eliminate crane interferences.

Among the above four cases, the crane scheduling in steelmaking and continuous casting process can be regarded as the most complex problem since the cranes need not only lift or lower heavy materials but also move them horizontally. Especially, when multiple cranes are employed, the potential collisions make this problem much more complex. In addition, Tanizaki *et al.* [14] divided the common track into small areas and at most two cranes are involved in each area, which is not in accordance with some practical production.

Unlike the existed research, this paper addresses the multiple-crane scheduling problem in steelmaking plants with a priority propagation, in which multiple objectives like productivity and smoothness of production are considered. With priority propagation based collision avoidance, the cranes are not restricted into a small area which is in accordance of real applications. The paper is organized as follows. Section 2 describes a motivating example in steelmaking plant and states. A strategy for collision avoidance named priority propagation is given in Section 3. On the ground of collision avoidance, a memetic algorithm is proposed in Section 4. Section 5 reports the experimental results, followed by conclusion.

2. Motivating Example of Crane Scheduling In Steelmaking Plants

This motivating example comes from a Chinese steelmaking plant, in which four processes are involved from steelmaking to continuous casting. As shown in Figure 1, after the steelmaking in LD converter and argon (AR) station, the molten steel in ladles is transported to the ladle furnace (LF) by cranes. Then it may be converted to RH vacuum furnace (RH) for further removing hydrogen and phosphorus, *etc.* Finally all ladles must be conveyed to the predefined continuous caster (CC). Note that, ladles belonging to one cast must be performed consecutively in exactly one caster without any interruption.

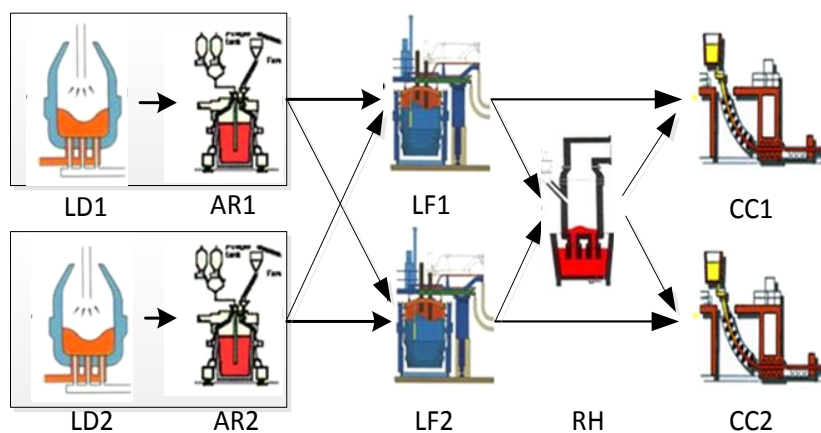


Figure 1. The Process from Steelmaking to Continuous Casting

Table 1. Plan and Schedule of 28 Tasks

| Task | ID. | Starting Station | Target Station | Time window | Task | ID. | Starting Station | Target Station | Time window |
|------|------|------------------|----------------|-----------------|------|------|------------------|----------------|-----------------|
| 1 | 2010 | LD1-AR1 | LF1 | 0:10:00-0:20:00 | 15 | 1050 | LD2-AR2 | LF2 | 2:40:00-2:50:00 |
| 2 | 8060 | LF2 | CC2 | 0:35:00-0:41:40 | 16 | 2020 | RH | CC1 | 2:45:00-2:50:00 |
| 3 | 1020 | LD2-AR2 | LF2 | 0:40:00-0:50:00 | 17 | 2050 | LD1-AR1 | LF1 | 2:55:00-3:06:40 |
| 4 | 2020 | LD1-AR1 | LF1 | 0:50:00-1:00:00 | 18 | 2040 | LF1 | RH | 3:15:00-3:20:00 |
| 5 | 2010 | LF1 | RH | 1:10:00-1:20:00 | 19 | 2030 | RH | CC1 | 3:25:00-3:36:40 |
| 6 | 1030 | LD2-AR2 | LF2 | 1:20:00-1:30:00 | 20 | 1030 | LF2 | CC2 | 3:35:00-3:46:40 |
| 7 | 2030 | LD1-AR1 | LF1 | 1:30:00-1:40:00 | 21 | 2060 | LD1-AR1 | LF1 | 3:35:00-3:50:00 |
| 8 | 1010 | LF2 | CC2 | 1:35:00-1:45:00 | 22 | 2050 | LF1 | RH | 3:55:00-4:10:00 |
| 9 | 2020 | LF1 | RH | 1:45:00-2:00:00 | 23 | 2040 | RH | CC1 | 4:05:00-4:13:20 |
| 10 | 1040 | LD2-AR2 | LF2 | 2:00:00-2:08:20 | 24 | 1060 | LD2-AR2 | LF2 | 4:25:00-4:33:20 |
| 11 | 2010 | RH | CC1 | 2:10:00-2:16:40 | 25 | 1040 | LF2 | CC2 | 4:35:00-4:50:02 |
| 12 | 2040 | LD1-AR1 | LF1 | 2:15:00-2:25:00 | 26 | 2060 | LF1 | RH | 4:35:00-4:51:40 |
| 13 | 2030 | LF1 | RH | 2:30:00-2:40:00 | 27 | 2070 | LD1-AR1 | LF1 | 4:40:00-4:50:00 |
| 14 | 1020 | LF2 | CC2 | 2:35:00-2:45:00 | 28 | 2050 | RH | CC1 | 5:00:00-5:11:40 |

According to the production schedule of the plant, we can obtain some information for further crane scheduling as represented in Table 1. The first part is about the material flow. For example, the ladle with ID number of 2010 starts its first process in LD1-AR1. It is transported to LF1 as task 1 and then to RH for refining as task 5. Finally, it is transported to CC1 for continuous casting as task 11. The second part is about the intensified limitation on continuous casting. We can find that the ladles with the ID number of 2010, 2020, 2030 and 2040 belong to the same cast and will be performed in CC1 consecutively. Similarly, ladles of 1020, 1030 and 1040 will be casted continuously and sequentially in CC2 as one cast. Thus, our duty in crane scheduling is to make decisions so that all these 28 conveyance tasks can be completed in the given time window. For instance, the task 6 (ID: 1030) can be started after its release time 1:20:00 at starting station LD2-AR2 and should be completed before its deadline 1:30:00 at target station LF2. Note that, a task ideally starts at its release time, but the movements of cranes on a common track may delay the start. It is also worthy pointing out that the serial number of tasks is accordance with the non-decreasing order of the release times.

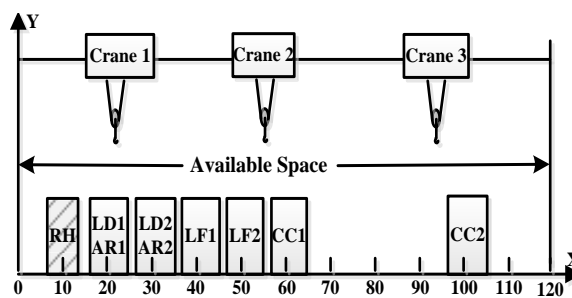


Figure 2. The Projection of Workshop Layout

From the perspective of production processes, all ladles should be conveyed from steelmaking to continuous casting. In other words, all materials should be transported from the left to the right as shown in Figure 1. However, the real production is different owing to the layout in the workshop. As shown in Figure 2, the RH vacuum furnace was built on the left of argon stations due to the limitation of space, causing unavoidable disorders in material flow. Meanwhile, three cranes

are running on the common track in the predefined space, implying that the left crane should stay on the left. If two adjacent cranes moving toward the opposite directions appear to be close to each other, crane interference occurs which is depicted in Figure 3. To avoid collisions, safety distance between two cranes is demanded.

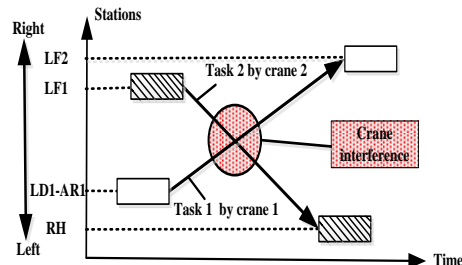


Figure 3. An Example of Crane Interference

For safety, all cranes run in two statuses: active transport and passive transport. Active transport means that a crane moves directly to its target. Passive transport denotes that a crane has to change its direction and move in an opposite direction of its target, which of course decreases the smoothness level of the crane operation. Thus, to smooth the crane operations, both the total transport distance and the total passive transport distance need to be minimized.

Therefore, solving the multiple-crane scheduling problem in steelmaking plants refers to making decisions about which crane, when and where to implement the given conveyance tasks under the constraints of safety distance and time window. The assignment of cranes to tasks and the real start time of all tasks need to be specified. At the meantime, the space-time trajectories of the cranes need to be determined so as to ensure the security among multiple cranes. Regarding optimization objectives, multiple goals are involved, including maximizing the total completed tasks, minimizing the total transport distance and the total passive transport distance. Meanwhile, the intensified limitation of all ladles in one cast on continuous casting process is expected to be treated specially.

3. Priority Propagation Based Collision Avoidance

Every crane operation comprises four stages. The first is to set up by moving from the current position to the starting station. The second is to lift the ladles at the starting station. The third is to convey ladles from the starting station to the target station. The last is to lower the ladles at the target station. Cranes may run with or without ladles. And the ladles may be fulfilled with the high-temperature molten steel or empty. To distinguish the importance of each task, all tasks are prioritized. Under the circumstances of crane interference, the relevant conveyance tasks will be sequenced according to the task priorities. This section demonstrates techniques in handling priorities and avoiding collisions.

3.1. Priority Initialization

In light of the importance of the production operation in steelmaking plants, several impact factors are considered when initializing the priorities. The first is about material properties. If a task is to convey the ladles with high-temperature molten steel, it must be endowed with a high priority for safety. The second focuses on operation processes. If a crane is lifting or lowering a ladle, any interruption may lead to a deadly danger. And the

conveyance tasks from the refinery (LF or RH) to the continuous caster (CC) also need to be emphasized so that all ladles in a cast can be performed consecutively and continuously. Thus, the priorities of tasks are classified into five categories as shown in Table 2. Note that, each task may comprise four or five steps of Table 2, and the priority of each task will be modified in time accordingly.

Table 2. Five Categories of Task Priorities

| Category | Motion mode | Priority |
|----------|------------------------------------|----------|
| 1 | Moving without a ladle | 1 |
| 2 | Moving with an empty ladle | 10 |
| 3 | Moving a ladle to LF or RH | 20 |
| 4 | Moving a ladle from LF or RH to CC | 50 |
| 5 | Lifting or lowering a ladle | 100 |

3.2. Priority Adjustment

For the continuity of the production, the less remaining conveyance times a task has, the earlier it should be implemented. Thus, on the ground of Table 2, the priorities are rectified according to the following adjustment rule.

Step 1: At time point t , create a task set for each category, determine which set the tasks are in, initiate their priorities based on the category and calculate the remaining time.

Step 2: Sequence all tasks in the task set of each category in non-decreasing order of the remaining times.

Step 3: Calculate the priority of each task by adding the sequence number in the task set they belong to, and then update.

3.3. Priority Propagation Based Collision Avoidance

The priority propagation is designed to avoid the potential collisions between any two adjacent cranes. It works as long as two cranes get close to the safety distance. The crane with higher value of priorities will stay active and move towards its original target. However, the crane with a relatively lower priority will be enforced to move passively in an opposite direction so as to avoid collisions. Besides, its priority value must be modified accordingly, just like one crane propagates its priority value to another.

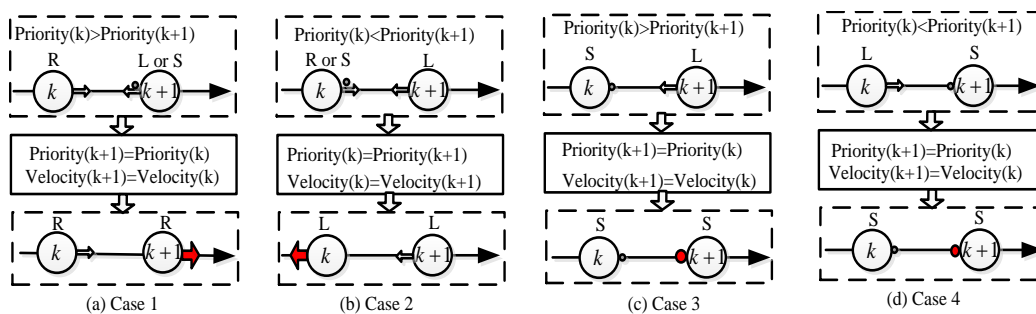


Figure 4. Four Cases of Priority Propagation

To describe the priority propagation mechanism clearly, we introduce three signals (L, R and S) to denote that a crane runs towards the left or right or stops moving, respectively. Considering the value of priorities, we can obtain four cases of priority propagation in Figure 4 under the condition that the distance between two cranes approaches the safety distance. In case 1 or 2, the crane with lower value of priorities is enforced to move towards an opposite direction. And in case 3 or 4, the crane with lower value of priorities has to wait in situ.

However, when three or more cranes are involved, the priority propagation needs to repeat multiple times till no interference exists between any two adjacent cranes. Thus, on the basis of the priority initialization, adjustment and propagation, the procedure of collision avoidance among all cranes is demonstrated as Figure 5. Note that, $position(k)$ denotes the current position of the crane k . Besides, if the returned results (e.g. 3L, 4R, 2S) from the forward and backward propagation steps are same, then no interference exists and the procedure stops.

```

Begin
    Step1: For K= 1 to (N-1) do                                %Forward propagation;
        If (position(k+1)-position(k)) < safety distance
            Propagate priority and change velocity ;
        Else Continue next loop;
        End if
    End for and return the number of (L,R,S);
    Step2: For K= (N-1) to 1 do                                %Backward propagation;
        If (position(k+1)-position(k)) < safety distance
            Propagate priority and change velocity ;
        Else Continue next loop;
        End if
    End for and return the number of (L,R,S);
    Step3: If The results of step 1 and 2 are equivalent        %Evaluation criterion;
        Output the result and update with the initial priority value;
    Else Return to step 1.
    End if
    
```

Figure 5. Priority Propagation Based Collision Avoidance

4. Memetic Algorithm Based Crane Scheduling

The memetic algorithm (MA), also called the hybrid genetic algorithm (hybrid GA), is one of the recent growing evolutionary algorithms. It was proposed by Moscato [15] and inspired by the social and cultural evolution. In MA, local searches as gradient-ascent techniques are imbedded to replace mutation operator, exploiting the search space and increasing the search intensity. At the same time, population-based approaches as evolutionary techniques explore the search space broadly, maintaining the diversification of solutions. As a result, the MA has proved quite effective in solving complex problems like multimodal function optimization [16-19].

We solve the multiple-crane scheduling problem in two steps. The first step is to assign the conveyance tasks to the cranes. The second is to derive the space-time trajectories of all cranes without interference. Thus, in this section, the techniques for dealing with tasks in setup stage are represented at first. Then the two step: encoding and decoding are represented. And finally the crossover and local search operators are designed.

4.1. Setup Stage of Tasks

After finishing one conveyance task, a crane must be setup for the next task by moving to its starting station. Here, two mechanisms can be chosen. The first is called “wait and see”. The crane waits at the current station till the last moment and then moves to the next starting station. The second is to “move and wait”. The crane moves to the next starting station immediately and then wait in situ. Note that, “wait and see” may result in the delay of the tasks while “move and wait” may lead to valueless movement, because arriving at

the station earlier than the schedule may cause valueless passive transport distances. As shown in Figure 6, with “move and wait”, crane 2 moves immediately towards the target station at first and returns for giving way to crane 2, which results in valueless movement.

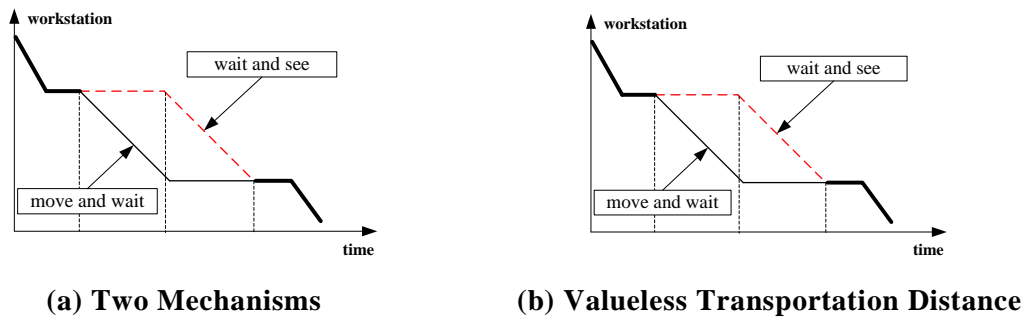


Figure 6. Setup for the Next Task

To achieve two goals of increasing the number of the completed tasks and reducing valueless movement simultaneously, equation (1) is utilized to control the start time for the setup stage of tasks. Where, t_i^{rs} is the actual start time that a crane moves to the next starting station, t_i^b is the release time of the task i , W_i^s is the starting station, P_t^k is the position of the crane k at time point t , v is the velocity. The setup parameter $\zeta (\zeta > 1)$ is employed to ensure that a crane can arrive at its target station on time.

$$t_i^{rs} = t_i^b - \frac{|P_t^k - W_i^s|}{v} \zeta \quad (1)$$

4.2. Encoding and Decoding

The crane assignment can be represented by a chromosome. The length of the chromosome is equal to the total number of tasks. The integer in each gene refers to the serial number of the allocated cranes. The position of genes indicates the serial number of tasks. Since the serial number of tasks is decided according to the non-decreasing order of the release times, all tasks allocated to the same crane can be performed in accordance with the serial number. Taking the code {2,1,3,2,2,1,2,3,3,1,2} as an example, when 11 tasks are assigned to 3 cranes, task (2,6,10) are allocated to crane 1, tasks (1,4,5,7,11) to crane 2, and task (3,8,9) to crane 3.

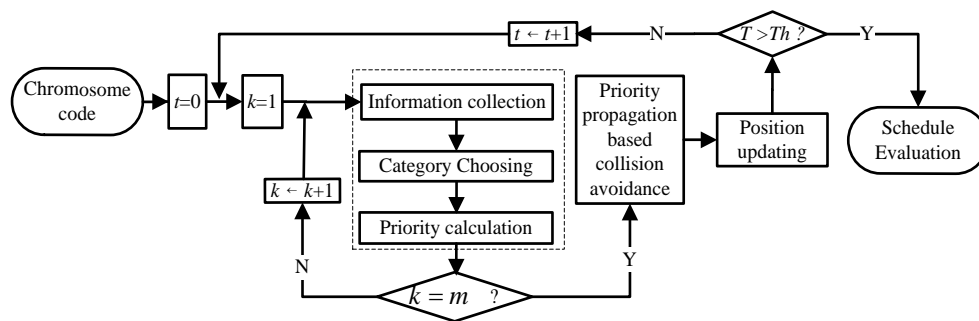


Figure 7. Priority Propagation Based Decoding Process

The key to obtain a feasible solution is to decode the chromosome with priority propagation based collision avoidance mechanism. Thus we should know the priority values in advance at each time unit, or make clear which category each task belongs to.

Therefore, the following information should be collected. The first part originates from the cranes, including the position, the serial numbers of the tasks currently handled on these cranes. The second is about these tasks in progress, like release times, deadlines, starting stations and target stations, and core information about which stage each task is attached to at present, setup, lifting, moving or lowering. Under the assistance of priority propagation based collision avoidance mechanism, the directions and velocities of all cranes are determined. Thus, the position of all cranes in the next time unit will be updated. Repeating this process till the termination time Th , the space-time trajectories for all cranes with guaranteed feasibility can be obtained. This decoding process is depicted in Figure 7.

4.3. Crossover and Local Search

Global search and local search are two key components in a meta-heuristic algorithm and the interaction between them can have significant impact on the efficiency of the algorithm. For global diversification, crossover operator is implemented by employing the two-point crossover method, in which two insertion points are randomly selected and the heads and tails of two parental chromosomes are exchanged. For local intensification, two local search operators are applied: swap and insertion. In the swap operator, two neighboring genes are selected randomly and then inter-exchanged. In insertion operator, a segment with the given number of genes which is selected randomly will be reinserted back into any other positions of the chromosome.

The crossover operator is applied for most of the individuals selected by probability, and two offspring will replace the parents absolutely. However, local search operators are used for each individual, and the individual can be updated only if a better solution is obtained. Moreover, local search operators can begin only after the completion of the crossover operator so that better results can be explored along with a large extent of diversification.

5. Computational Experiments

5.1. Parameter Calibration

As we know, the crossover operation, the local search operation and the population size are the three factors that affect the performance of MA. To calibrating all those factors, we design four levels for each of those factors. That means we have kinds of combination for MA. The detailed scheme is listed as follows.

⊙Local search: 4 levels (Mutation, Insertion, Swap, Swap + Insertion)

⊙Crossover: 4 Levels (Single-point crossover, two-point crossover, uniform crossover, uniform two-point crossover)

⊙Population: 4 levels (10, 20, 30, 40)

In order to be more precise, each combination of MA is run under 5 cases. The proposed algorithm is encoded with C++ and runs in visual studio 2015 environment. The common stopping criterion is CPU time which we set to 100s for each combination and case. To sum up, we have $64 \times 5 = 320$ treatments. Here, the weight coefficients of three objectives ($\lambda_1, \lambda_2, \lambda_3$): the coefficient of the total completed tasks(CT), the coefficient of total transport distance(TD) and the coefficient of total passive transport distance(PD), are set as 100000, 1000 and 1 respectively and briefly.

Once all the experiments are done, the relative percentage deviation (RPD) is proposed to measure the solution, which is denoted as $RPD = (Some_{sol} - Best_{sol}) / Best_{sol} \times 100$. $Some_{sol}$ means the solution by one combination and $Best_{sol}$ is the best fitness for the same instance. In order to identify the factors, the analysis of variance (ANOVA) technique is proposed as a parametric statistical technique. For ANOVA, three main hypotheses must

be satisfied, namely the independence of the residuals, homogeneity of variance and normality of the residuals. The obtained results about RPD values are reported in Table 3 and mean plot for three factors (local search, crossover and population size) is shown in Figure 8.

The P-Value in Table 3 shows that three factors have significant impact on the performance of the MA within a 95% confidence interval.

Table 3. The Results of RPD Values

| Source | Sum of Squares | Df | Mean Square | F-Ratio | P-Value |
|--------------|----------------|----|-------------|---------|---------|
| Local search | 781.793 | 3 | 260.598 | 9.006 | 0.012 |
| Crossover | 528.520 | 3 | 176.173 | 6.089 | 0.030 |
| Population | 409.541 | 3 | 136.514 | 4.718 | 0.051 |

As you can see in Figure 8(a), the insertion operation is obviously better than the two other local search operations: mutation and swap. And when an insertion operation is combined with the swap operation, the performance of the MA is further improved. We can also see in Figure 8(b) that the uniform crossover obtains a worst performance in the proposed MA and the two-point crossover obtains a relatively good performance. About the scale of population, a size of 10 is the best choice based on the Figure 8(c). Thus, in the following experiments, we utilize the swap and insertion as the local search and the two-point crossover as the crossover operation, and the population size is set to 10.

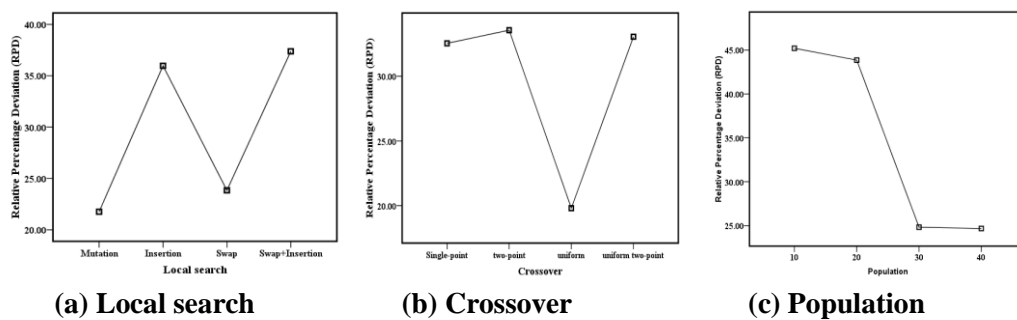


Figure 8. Mean Plots for the Factors of MA ANOVA Calibration Experiment

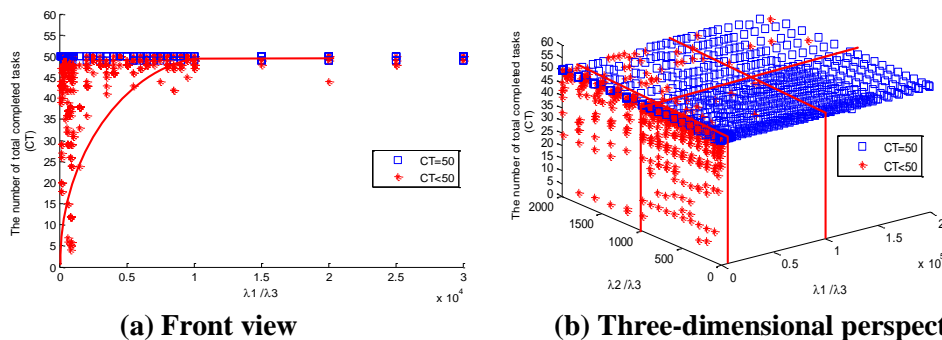


Figure 9. Influence of Weight Coefficients on the Objective – The Total Completed Tasks (CT)

The influence of three weight coefficients ($\lambda_1, \lambda_2, \lambda_3$) in objectives also needs to be calibrated. Since the primary objective is maximizing the number of total completed tasks, we mainly focus on that. It shows clear in Figure 9(a) when λ_1/λ_3 is less than 10^4 ,

most of 50 conveyance tasks cannot be completed on time. If λ_1/λ_3 is larger than 10^4 , whether a conveyance task can be completed or not is dependent on other coefficients. From Figure 9(b) we know that the larger the ration λ_2/λ_3 , the less the likelihood that all tasks will be completed. Meanwhile, a reasonable interval is revealed. If $\lambda_1/\lambda_3 \geq 10^5$ and $\lambda_2/\lambda_3 \leq 10^3$, all tasks can be completed on time.

5.2. Revisited Motivating Example

The motivating example in section 2 is revisited. We set λ_1, λ_2 and λ_3 as 100000, 1000 and 1 respectively. Both lifting and lowering time are 60s. The safety distance is 3m, the setup parameter is 1.5, the time unit is 1s, and the crane velocity 0.5m/s. Using the proposed algorithm in which priority propagation and memetic algorithm are hybridized together, the computational results about crane assignment, start time and completion time of all tasks are obtained and reported in Table 4. And the space-time trajectories of all cranes are depicted in Figure 10.

Table 4. The Schedule for 28 Tasks of the Motivating Example

| Task | ID. | Crane Assignment | Start-Completion time | Task | ID. | Crane Assignment | Start-Completion time |
|------|------|------------------|-----------------------|------|------|------------------|-----------------------|
| 1 | 2010 | 2 | 0:10:01-0:12:10 | 15 | 1050 | 2 | 2:40:01-2:42:10 |
| 2 | 8060 | 1 | 0:35:00-0:39:09 | 16 | 2020 | 2 | 2:45:50-2:49:59 |
| 3 | 1020 | 2 | 0:40:01-0:42:10 | 17 | 2050 | 3 | 2:55:00-2:57:09 |
| 4 | 2020 | 3 | 0:50:01-0:52:10 | 18 | 2040 | 3 | 3:15:00-3:17:49 |
| 5 | 2010 | 3 | 1:10:00-1:12:49 | 19 | 2030 | 2 | 3:25:39-3:29:48 |
| 6 | 1030 | 3 | 1:20:01-1:22:10 | 20 | 1030 | 1 | 3:35:39-3:39:48 |
| 7 | 2030 | 3 | 1:30:01-1:32:10 | 21 | 2060 | 2 | 3:35:01-3:37:10 |
| 8 | 1010 | 1 | 1:35:39-1:39:48 | 22 | 2050 | 2 | 3:55:00-3:57:49 |
| 9 | 2020 | 3 | 1:45:00-1:47:49 | 23 | 2040 | 2 | 4:05:00-4:09:09 |
| 10 | 1040 | 2 | 2:00:01-2:02:10 | 24 | 1060 | 3 | 4:25:01-4:27:10 |
| 11 | 2010 | 2 | 2:10:01-2:14:10 | 25 | 1040 | 2 | 4:35:01-4:39:10 |
| 12 | 2040 | 2 | 2:17:50-2:19:59 | 26 | 2060 | 3 | 4:35:01-4:37:52 |
| 13 | 2030 | 2 | 2:30:00-2:32:49 | 27 | 2070 | 3 | 4:40:02-4:42:51 |
| 14 | 1020 | 1 | 2:35:39-2:39:48 | 28 | 2050 | 3 | 5:00:01-5:04:10 |

Comparing the planned time window in Table 1 and real start-completion time in Table 4, it can be concluded that all the 28 tasks have been completed in the given time window. However, even for the most senior dispatchers, at most 4 tasks can be handled concurrently, which are far less effective than the proposed algorithm.

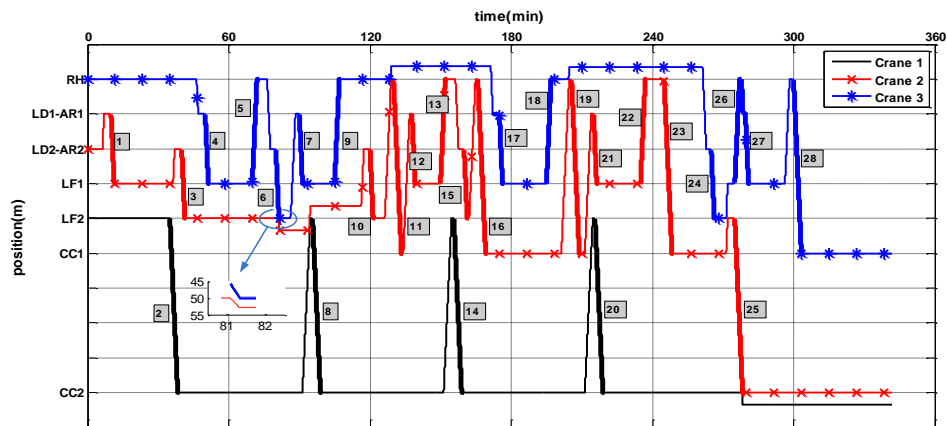


Figure 10. Space-time Trajectories of Three Cranes

In Figure 10, crane 1 always works from CC2 to at most LF2, crane 3 keeps transporting from RH to at most CC1, but crane 2 closes to crane 1 or 3 from time to time. This phenomenon is exactly the same as that in real production. And further investigation can reveal the fact that no interference between any two neighboring cranes exist. Even

when crane 1 and 2 are quite close at 81min, the distance between them is larger than the safety distance. It demonstrates that the schedule of the 28 tasks in Table 3 and the trajectories of 3 cranes in Figure 10 are quite reasonable. And hence the proposed algorithm via the combination of priority propagation and memetic algorithm is effective.

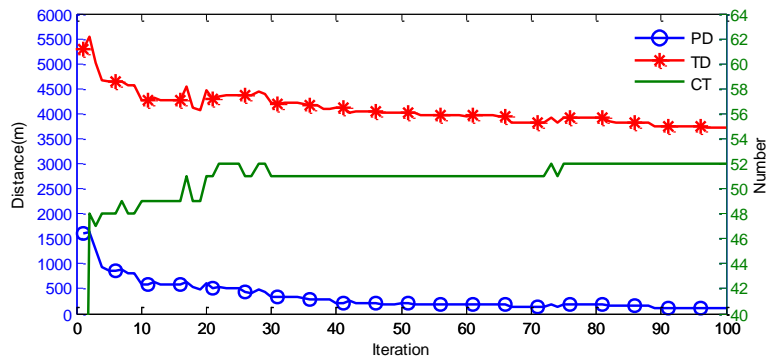


Figure 11. Convergence Processes of Three Objectives

To test the computational efficiency of the proposed algorithm, a larger case with 52 tasks also from the Chinese steelmaking plant of motivating example is applied. The allowed iteration time is 100 and the result is presented in Figure 11. It shows clearly that the number of the total completed tasks (CT) reaches 52 at 75th iteration. Both the total transport distance (TD) and total passive transport distance (PD) reduce gradually and largely remain unchanged in the late stage of iterations. It proves that the proposed algorithm has rapid convergence speed and superior global optimization ability.

6. Conclusion

This paper solves the multiple-crane scheduling problem in steelmaking plants with three objectives using a memetic algorithm with priority propagation. In the priority propagation, the tasks are divided into 5 categories and then prioritized based on the material properties and operation processes. In the case of crane interference, these priorities are propagated so that potential collisions are eliminated.

In addition, the crossover operators and local search methods in the memetic algorithm are compared and calibrated with analysis of variance technique to achieve the fine balance of global diversification and local intensification. Computational experiments on motivating examples from a Chinese steelmaking plant show that the proposed algorithm is effective and efficient. Further research can focus on crane rescheduling due to uncertain disturbance to guarantee a quick response.

Acknowledgments

This work is funded by the National Science Foundation of China under grants of 51275366, 50875190 and 51305311 and the Specialized Research Fund for the Doctoral Program of Higher Education of China (20134219110002). The authors thank the suggestions and comments by anonymous referees to help improve this paper.

References

- [1] K. H. Kim and Y. M. Park, "A crane scheduling method for port container terminals", *European Journal of Operational Research*, vol. 156, no. 3, (2004), pp. 752-768.
- [2] L. Moccia, J. F. Cordeau, M. Gaudioso and G. Laporte, "A branch-and-cut algorithm for the quay crane scheduling problem in a container terminal", *Naval Research Logistics*, vol. 53, no. 1, (2006), pp. 45-59.
- [3] M. Sammarra, J. F. Cordeau, G. Laporte and M. F. Monaco, "A tabu search heuristic for the quay crane scheduling problem", *Journal of Scheduling*, vol. 10, no. 4-5, (2007), pp. 327-336.
- [4] C. Bierwirth and F. Meisel, "A fast heuristic for quay crane scheduling with interference constraints", *Journal of Scheduling*, vol. 12, no. 4, (2009), pp. 345-360.
- [5] S.H. Chung and K.L. Choy, "A modified genetic algorithm for quay crane scheduling operations", *Expert Systems with Applications*, vol. 39, no. 4, (2012), pp. 4213-4221.
- [6] K. H. Kim and K. Y. Kim, "An optimal routing algorithm for a transfer crane in port container terminals", *Transportation Science*, vol. 33, no. 1, (1999), pp. 17-33.
- [7] K. Y. Kim and K. H. Kim, "Heuristic algorithms for routing yard-side equipment for minimizing loading times in container terminals", *Naval Research Logistics*, vol. 50, no. 5, (2003), pp. 498-514.
- [8] C. Zhang, Y. Wan, J. Liu and R. Linn, "Dynamic crane deployment in container storage yards", *Transportation Research Part B*, vol. 36, no. 6, (2002), pp. 537-555.
- [9] R. Linn and C. Zhang, "A heuristic for dynamic yard crane deployment in a container terminal", *IIE Transactions*, vol. 35, no. 2, (2003), pp. 161-174.
- [10] W. Yan, Y. Huang and J. He, "An investigation into dynamic yard crane deployment and comparisons between hill-climbing and best-first-search algorithms", *International Journal of Computer Applications in Technology*, vol. 32, no. 4, (2008), pp. 254-264.
- [11] A. Che and C. Chu, "Single-track multi-hoist scheduling problem: a collision-free resolution based on a branch-and-bound approach", *International Journal of Production Research*, vol. 42, no. 12, (2004), pp. 2435-2456.
- [12] Z. Zhou and J. Liu, "A heuristic algorithm for the two-hoist cyclic scheduling problem with overlapping hoist coverage ranges", *IIE Transactions*, vol. 40, no. 8, (2008), pp. 782-794.
- [13] X. Li, Felix T.S. Chan and S.H. Chung, "Optimal multi-degree cyclic scheduling of multiple robots without overlapping in robotic flowshops with parallel machines", *Journal of Manufacturing Systems*, vol. 36, (2015), pp. 62-75.
- [14] T. Tanizaki, T. Tamura, H. Sakai, Y. Takahashi and T. Imai, "A heuristic scheduling algorithm for steel-making process with crane handling", *Journal of the Operations Research Society of Japan*, vol. 49, no. 3, (2006), pp. 188-201.
- [15] P. Moscato, "An introduction to population approaches for optimization and hierarchical objective functions: A discussion on the role of Tabu Search", *Annals of Operations Research*, vol. 41, no. 2, (1993), pp. 85-121.
- [16] A. M. Acilar and A. Arslan, "A novel approach for designing adaptive fuzzy classifiers based on the combination of an artificial immune network and a memetic algorithm", *Information Sciences*, vol. 264, (2014), pp. 158-181.
- [17] A. Divsalar, P. Vansteenwegen, K. Sörensen and D. Cattrysse, "A memetic algorithm for the orienteering problem with hotel selection", *European Journal of Operational Research*, vol. 237, no. 1, (2014), pp. 29-49.
- [18] Y. Li, L. Jiao, P. Li and B. Wu, "A hybrid memetic algorithm for global optimization", *Neurocomputing*, vol. 134, (2014), pp. 132-139.
- [19] L. Ma, M. Gong, J. Liu, Q. Cai and L. Jiao, "Multi-level learning based memetic algorithm for community detection", *Applied Soft Computing*, vol. 19, (2014), pp. 121-133.

Authors



Yongnian Mao, he received the B.S. degree from Science and Technology College of Hubei University of Arts and Science in 2012. He is currently an MD-PhD in the Wuhan University of Science and Technology. His current research interests include intelligent manufacturing, evolutionary algorithms, and their application in industrial manufacture.