A simulation-based immune genetic algorithm for the crane scheduling problem

Huizhi Ren* and Jian Tao

School of Mechanical Engineering, Shenyang University of Technology, Shenyang, P. R. China

ABSTRACT

The paper studied a multi-machine and multi-task warehouse crane scheduling problem. The collision between adjacent cranes needs to be avoided in the scheduling problem. Considering the involved operation requirement and condition, a process simulation model was formulated to evaluate a candidate crane schedule through rehearsing the crane operation process. Based on the process simulation, an immune genetic algorithm was developed to iteratively improve the crane schedule. The process simulation-based immune genetic algorithm was tested on a group of practical instances, which showed that it was effective for the crane scheduling problem.

Key words: Immune genetic algorithm; process simulation; warehousing problem; crane scheduling

INTRODUCTION

Cranes are commonly used in workshops, storehouses and storage yards. According to the different installation occasions and structures, they can be divided into two types: gantry crane, moving with track and support [1], and bridge crane, running on storehouse walls or cross beam. The gantry crane is often used in ports, station yards and other outdoor occasions for loading and unloading goods while the bridge crane is used in workshops, storehouses for transporting work pieces and accessing the goods. In the warehouse, to overcome the equipment resource constraint [2], more than one cranes are usually installed on one track to deal with the frequent access operations. The potential collision between two moving cranes should be taken into consideration when making the crane operation schedule [3]. In the actual situation, the warehousing operation sequence is also a key constraint to be considered for the stacking level or storage/retrieval order [4].

The crane scheduling problem is to reasonably dispatch available cranes so as to finish the warehousing tasks effectively and efficiently under the premise of complying with the crane operation safety regulation. In daily warehousing management, reasonable and effective crane scheduling makes a great contribution to utilization improvement and cost control in storage [5]. For the multi-processor and multi-tasking characteristic and complex spatial and temporal constraints, the crane scheduling problem is a typical NP-hard problem.

There have been a lot of researches for crane scheduling problem. Dohn and Clausen [6] studied on operation scheduling of steel materials storage, related to crane scheduling. Han et al. [7] took a theoretical study about operation scheduling (order) in storage system, argued that the access to scheduling problem is a NP-hard problem, where the so-called storage/retrieval scheduling is same as the crane scheduling. Bozer et al [8] built a Chebyshev TSP model about crane operation scheduling and proved that the complexity of the problem is NP-hard. Koh et al. [9] found that round storage system, the operation of storage finished by a tower crane located in the center, access equipment moved in the direction of radius and circumference at the same time. This research built a crane travel time mathematic model, the model and the calculation of crane travel time have some reference value to this paper. Lerher [10] considered an automated storage/retrieval system and improved the system operating performance through the crane travel time model. Zheng et al. [11] gave a simulation model for crane scheduling in workshop.
Most of these researches used mathematics model and numerical optimization methods and thus were difficult to handle the complex spatial and temporal constraint and the uncertainty in actual situation.

Aiming at the practical requirements in warehousing crane optimal scheduling, this paper combined simulation method and heuristic optimal scheduling method and proposed a new immune genetic algorithm based on process simulation. In the algorithm, a simulation model abstractly described the operation starting and destination locations and operation style in actual situation storage, built a collision avoidance rules on the basis of crane operation safety regulation. The immune genetic algorithm was developed based on the simulation method to make up its weakness in optimization ability. It also provided a new idea to solving the complex crane scheduling problem.

THE PROCESS SIMULATION-BASED IMMUNE GENETIC ALGORITHM

The crane optimal scheduling problem considered in this paper takes place in a warehouse which stores bundles of steel pipes, as shown in Fig 1. The storage area is divided into a number of storage columns through the specially designed shelves. Each storage column can stack multiple bundles of pipes. In order to conveniently access and transport, the pipes have been bunched into the square bundles (as shown in Fig 1) before they are put into the storage. Due to the different sizes of pipes, the bundle can hold the different number of pipes. When bundles of pipes are put into storage, the crane slings them and moves them from the storage platform to a specified storage column. When retrieving some a bundle, the crane removes it to the exit area. If a pipe bundle is not on the top of its storage column, the above bundle needs to be moved aside in advance, which is called a passive shuffling operation. To feed up the storage/retrieval efficiency and keep bundles in order, the crane operation plan can also include some active shuffling operations which always aim to keep the pipes of same steel grade and sizes in the same column.

The crane scheduling problem in the storage is always triggered by a given operation plan which can include the storage/retrieval operations and active and passive shuffling operations. Most of the storage/retrieval operations are required to carry out in a given time window and some operations can be forced to executed in a logical sequence (for example the operations in the same storage column). The crane scheduling problem need to arrange the suitable crane to the required operations in order to execute these planed operations accurately and timely. The crane scheduling should also consider the security regulation constraint that the adjacent cranes need to keep a safety distance. Hence, crane can be forced to carry out some passive moves (include loading and no-loading passive moves), as shown in Fig 2. The proper crane scheduling can make sure finish tasks punctually, reduce the crane operation load and finish the warehousing operations as early as possible.
After determining the crane operation assignment and operations execution sequence of every crane, the crane operation process simulation model can be formulated according to the positional relationships of these crane tasks. The model can conveniently describe the crane operation process and thus ascertain tasks implementation to comply with the logical sequence and temporal constraints. The simulation model can also conveniently describe the crane spatial position in different time and thus judge whether there exists spatial conflict between the adjacent cranes. The space conflict is solved by using the given avoidance principle.

In order to get the feasible and relatively optimal crane schedule, the immune genetic algorithm is adopted to correct and change some parts of the crane operation assignments and operation orders based on the crane operation roadmap resulted from the process simulation. The immune genetic algorithm can search the solution pace for better crane dispatch and operation order through a series of iterations. Therefore, the core idea of the immune genetic algorithm based on process simulation is that the simulation model evaluate the crane schedule through the process simulation while the intelligent optimization algorithm generate random scheduling plan or superior plan through multiple iteration according to the different valuation feedback resulted from the process simulation.

SIMULATION MODEL

The simulation model is based on the given crane schedule, with objective to evaluate the crane schedule. It can first judge the feasibility and give a feedback value to the scheduling algorithm. The simulation model introduces the following symbol:

\( C \) : the set of cranes, \( C = \{1, 2, \ldots, M\} \)
\( \Omega \) : the set of operations, \( \Omega = \{1, 2, \ldots, N\} \)
\( \Psi \) : storage position set, \( \Psi = \{1, 2, \ldots, P\} \)
\( \Theta \) : range of the involved time intervals, \( \Theta = \{1, 2, \ldots, H\} \)
\( s_i, d_i \) : the starting and destination storage positions of operation \( i \), \( i \in \Omega \)
\( [t_i, T_i] \) : the permitted time window for carrying out operation \( i \), \( i \in \Omega \). For some operation, such as the active shuffling operation, there is no time window constraint and thus the time window is \([0, \infty]\).
\( \Gamma = \{ (o_1, o_1'), (o_2, o_2'), \ldots \} \) : the set of operation pairs subjected to the given sequence constraint, that is, operation \( i \) need to be finished before starting \( i' \) if \((i, i') \in \Gamma \).
\( b_i, e_i \) : the starting and finish time of operation \( i \) in the simulation, \( i \in \Omega \).
\( P_{c,i} \) : the position of crane \( c \) at time \( i \) in the simulation, \( i \in \Omega \).
\( L_{Min} \) : the permitted minimal distance between two adjacent cranes according to the related safety regulation.

Based on the above notations, the simulation model is formulated as follows:

\[
\text{Obj Min} \left( \text{Max} \, \frac{e_i}{\in \Omega} \right) \quad (1)
\]

s.t. \hspace{1cm}
\begin{align*}
 b_i & \geq t_i, \quad e_i \leq T_i \quad (2) \\
 e_i & < b_{i'} \text{ for } (i, i') \in \Gamma \quad (3)
\end{align*}
\[ P_{c,t} - P_{c-1,t} \geq L_{\text{Min}} \] (4)

As shown in formula (1), the objective is to minimize the latest finish time (make-span). Formula (2) is designed to satisfy the time window constraint, which ensures the operation must be executed in the given time window. Formula (3) ensures the sequence constraint and formula (4) keeps the safe distance between the adjacent cranes.

Fig. 3: Crane operation simulation

Requirements of crane operation need to be considered in simulation process:

1. To avoid crane spatial conflict. Adjacent cranes cannot move across each other and should keep wide berth.
2. To avoid task temporal conflict. Tasks assigned to the same crane need to be executed one by one and cranes need to run empty between the successive tasks, which means that empty run from the objective location of former task to the beginning location of next one.

Claim (1) can result in passive movement, which means crane is driven by the neighborhood crane. As shown in Fig. 2. The following principles should be concerned to dealing with crane spatial conflict:

1) Empty cranes avoid the loaded crane;
2) Tasks with the strict time window constraint is prior. In fact, the storage/retrieval operation always has more strict time window constraint and thus should be assigned a higher priority.

Simulation model reflects not only the characteristic of crane operation and collision avoidance principle, but also the property of crane and its operation. The crane property includes spatial position (described by storage location under the crane), load style, direction of movement, the starting position, moving distance, operation time and the given operation order. The operation property includes location of start, operation style, time window, precursor operation and so on.
Aiming at the crane scheduling plan given, simulation process is shown in Figure 3. Simulation clock propels as a fixed increment, successively execute operation according to the given crane operation order. Every clock cycle should test the existing of crane spatial conflict, and dispose the detected conflict as the certain processing rules of conflict. The running crane is paused in order to dispose the conflict (the crane may be moved passively), even the simulation clock will be rolled back to delay some operation.

THE IMMUNE GENETIC ALGORITHM

The genetic algorithm (GA) has been applied to various optimization problems\textsuperscript{[12]}. The algorithm has a good universality, i.e., the GA approach is independent on the particular problem structure and model. The immune genetic algorithm is a new variant optimization method of the GA. It utilizes the functions of antibody concentration selection and clone proliferation in artificial immunology to make up the weakness of local search ability and low efficiency of global search that typically exists in the pure genetic algorithm\textsuperscript{[13]}. In the early stage of the population evolution, chromosomes of high affinity can be chosen through the high frequency variation and clone selection and thus the immune genetic algorithm improve its local search ability. In the late stage of population evolution, because of balance of evolution and quick recession vital power of some memory cells, a large number of new memory cells are generated which provides the population higher diversity and wider searching zone and avoids premature convergence (precocity). That is to say the algorithm has higher efficiency in global search.

In crane scheduling, the make-span (finish time of the last operation) of feasible crane schedule, which ensure all tasks to be finished on time, is taken as the objective function value. For the infeasible schedule, the objective function value is viewed as a very big number $Q$ (greater than all possible feasible schedules). Antibody is the solution of the corresponding antigen, so its gene sequence is equal to the combination of crane operation sequence. The fitness (affinity) function is defined as $F = Q - \text{Makespan}$. In this way, the corresponding fitness of infeasible plan is 0 and $F$ is made to describe the merits and demerits of different crane schedules. Basic flow of the immune genetic algorithm as following:

1) Problem recognition: verify the operation tasks to schedule according to the given operation plan.
2) Initialize antibody population. Antibody population is composed of a series of antibodies. The antibody is encoded in an integer list. For example, there are 2 cranes (crane 1 and 2) and 6 operations (operation 1, 2, ..., 6), the combination [2, 4, 6; 1, 3, 5] is an antibody code. In our implemented initialization, the antibody is formed with random crane dispatch and the operations of the same crane are executed according to their time window and logical sequence or take random order when some operations have the common time window and no logical sequence constraint.
3) Calculate the affinity of antibody-antigen. Encode the antibody code into a crane schedule and put into the designed process simulation and the affinity can be computed based on the feedback value from the simulation. All antibody-antigen in the population need to be simulated and assigned the affinity values.
4) Calculate antibody concentration. The antibody concentration is based on the relationship of antibodies in the population and can be calculated through formula (5).

$$\text{Density}_i = \frac{\sum_{j=1}^{N_{Ab}} \text{count}_{ij}}{N_{Ab}}$$  \hspace{1cm} (5)

where $\text{count}_{ij} = \sum_{k=1}^{N_g} Ab_{ij}^k / N_g$, $Ab_{ij}^k$ is 0-1 integer. $\text{Density}_i$ is the antibody concentration for antibody $i$ and $\text{count}_{ij}$ is the similarity degree of antibody $i$ and antibody $j$, $Ab_{ij}^k$ is the similarity sign number in the $k$th gene for antibody $i$ and antibody $j$, $N_{Ab}$ is the number of antibodies and $N_g$ is the number of antibody genes.

5) Selection of antibody. Adopt the wheel selection to select a few antibodies of high affinity to join the new antibody population (elite strategy).
6) Antibody diversification. The antibody with high affinity and low concentration is chosen for crossover (two-point crossover method) and mutate (insertion mutation method), and the produced consequence is append into the new antibody population.
7) If the number of new antibodies has arrived the antibody population size but the convergence condition has not been satisfied, begin the next iteration from the first step.

SIMULATION EXPERIMENTS AND RESULT

In order to verify effectiveness of the proposed crane scheduling approach combining the process simulation and
immune genetic algorithm, we complement the approach under the environment of VC++ 2010. And numerical experiments are run on the data from a practical pipe warehouse whose layout is shown in Fig. 1.

Ten problem instances of different scales have been chosen and disposed in actual operation and taken as experimental data. The crane operation time is set as following: 1 unit (about 1 second in fact) is the time of crane moving one storage column in storage area; the time for executing one lift or laying down is 20 units. In test, the scale of antibody population is set to 30, crossover rate is set to 0.60 and mutation rate is set to 0.15. The computation time is limit in 300 CPU-seconds for each problem instance and the optimization process will be forced to terminate after 300 seconds even if the convergence condition has not been satisfied.

**TABLE 1. The experiment result**

<table>
<thead>
<tr>
<th>Problem</th>
<th>Scale</th>
<th>Obj</th>
<th>Idle ratio</th>
<th>Iters</th>
<th>Sols</th>
<th>CPUs</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>27</td>
<td>2163</td>
<td>19.3%</td>
<td>17</td>
<td>13</td>
<td>143</td>
</tr>
<tr>
<td>2</td>
<td>29</td>
<td>2077</td>
<td>16.7%</td>
<td>23</td>
<td>11</td>
<td>213</td>
</tr>
<tr>
<td>3</td>
<td>33</td>
<td>2316</td>
<td>14.8%</td>
<td>37</td>
<td>14</td>
<td>300*</td>
</tr>
<tr>
<td>4</td>
<td>34</td>
<td>2204</td>
<td>15.1%</td>
<td>13</td>
<td>31</td>
<td>123</td>
</tr>
<tr>
<td>5</td>
<td>41</td>
<td>2646</td>
<td>13.3%</td>
<td>27</td>
<td>9</td>
<td>300*</td>
</tr>
<tr>
<td>6</td>
<td>57</td>
<td>3484</td>
<td>9.5%</td>
<td>13</td>
<td>23</td>
<td>261</td>
</tr>
<tr>
<td>7</td>
<td>57</td>
<td>3790</td>
<td>10.6%</td>
<td>14</td>
<td>85</td>
<td>274</td>
</tr>
<tr>
<td>8</td>
<td>59</td>
<td>2915</td>
<td>4.3%</td>
<td>18</td>
<td>6</td>
<td>300*</td>
</tr>
<tr>
<td>9</td>
<td>61</td>
<td>4158</td>
<td>11.1%</td>
<td>19</td>
<td>17</td>
<td>300*</td>
</tr>
<tr>
<td>10</td>
<td>63</td>
<td>3901</td>
<td>8.7%</td>
<td>12</td>
<td>29</td>
<td>209</td>
</tr>
</tbody>
</table>

Table 1 shows the experiment results for the ten problem instances where the problem size (number of operations to execute), objective value (make-span), the average crane idle ratio, iteration number, number of feasible solutions produced in the iterations and computing time (CPU seconds) is denoted by Scale, Obj, Idle, Iters, Sols and CPUs, respectively. The italic characters in CPUs mean the problem instance terminated for the computing time limit instead of meeting the termination condition in the iteration step.

The simulation experiment result shows that the simulation model and immune genetic algorithm are both effective and the proposed crane scheduling approach, based on the combination of the process simulation and intelligent optimization, can work out a satisfactory crane schedule in the acceptable computing times.

**CONCLUSION**

The multi-machine and multi-task crane scheduling problem in a pipe warehouse was studied and a solution approach combining process simulation and immune genetic algorithm was proposed. The immune genetic algorithm produced and updated an antibody population while the simulation model can evaluate the antibodies through rehearsing the crane operating process. The corresponding program was also implemented and tested on a series of problem instances. The numeric experiments showed that the proposed solution approach was effective which could work out a relatively optimized crane schedule in the acceptable computing times. Against different problem scales, the solution quality is relatively stabilizing and the computing time is acceptable.

**Acknowledgments**

This research is partly supported by National Natural Science Foundation of China (71201105), China Postdoctoral Science Foundation (2013M530947) and general Scientific Research Project of the Education Department of Liaoning Province (Grant No. L2014480).

**REFERENCES**