Optimization of yard crane scheduling using particle swarm optimization with genetic algorithm operators (PSOGAO)

M Manoj Kumar and S N Omkar*

Department of Aerospace Engineering, Indian Institute of Science, Bangalore 560 012

Received 12 June 2007; revised 13 February 2008; accepted 21 February 2008

This study presents optimization of a yard crane schedule (YCS) to perform a given set of container handling jobs with different ready times in order to minimize the sum of job waiting times or the total completion time for handling job. Particle swarm optimization with genetic algorithm operators (PSOGAO) has been successfully applied to optimize YCS. PSOGAO has been observed more efficacious than heuristic method.

Keywords: Container terminal, GA operators, Heuristics, Particle swarm optimization, Yard crane scheduling

Introduction

Yard cranes are commonly used to load containers onto or unload containers from trucks in land-constrained port container terminals. In mega port container terminals, it is important to reduce waiting time of trucks by determining a good schedule for a yard crane in each yard zone to handle containers for trucks arriving at different times. Currently, human judgment and simple heuristics1 are used to solve the problem of determining a yard crane schedule (YCS) in a yard zone to handle containers for a set of trucks (jobs) arriving at different times, which minimizes the sum of the truck waiting times. Problems associated with dispatching and scheduling vehicles and handling equipment arise frequently in logistics systems2. Atkins3 presented details of landside operations including comparisons of port container handling equipment and storage yard strategies. Research has focused on increasing throughput of a port container terminal and, in particular, on decreasing turnaround times of ships4. Bish et al5 studied the problem of assigning each discharging container to a yard location and assigning dispatching vehicles to containers so as to minimize the time to discharge all containers from a ship.

Particle swarm optimization (PSO), an evolutionary computation technique6-7, is a population based optimization tool. Original PSO is different from other evolutionary-type methods in a way that it does not use filtering operation to generate a new generation (crossover and mutation operators) and members of entire population are maintained throughout the search procedure so that information is socially shared among individuals to direct the search towards the best position in the search space. In recent years, a lot of studies8-13 has focused on variant of PSO.

In this paper, particle swarm optimization with genetic algorithm operators (PSOGAO) is presented and successfully applied to an YCS problem.

Mathematical Model

YCS problem is formulated as an integer programming optimization. There are n jobs (trucks) in the zone to be handled by yard crane in the current planning period. Let \( r_i, i = 1, 2, \ldots, n \), be the ready time (truck arrival time) of job \( i \); \( h_i, i = 1, 2, \ldots, n \), be the time required by the yard crane to handle job \( i \); and \( d_{ij}, i = 1, 2, \ldots, n \) and \( j = 1, 2, \ldots, n \), be the time required for the yard crane to travel from the location of job \( i \) to the location of job \( j \). The \( n \) jobs are indexed in such a way that \( r_i, d_i = r_{i+1} \). The following decision variables...
are needed to formulate YCS problem mathematically: $t_i$, the time at which yard crane completes handling of job $i$ and

$$\begin{align*}
X &= \begin{cases} 
1 & \text{if job } i \text{ is handled before job } j \\
0 & \text{otherwise}
\end{cases}
\end{align*}$$

Denote the set of $X_i$ by $X$. For the truck dispatched to job $i$, its waiting time in the yard is given by $t_i - h_i - r_i$. With given $r_i$, the problem of finding the optimal schedule that minimizes the sum of job waiting times can be stated as

$$\text{Minimize } \sum_{i=1}^{n} (t_i - h_i - r_i)$$

subject to

$$t_i - r_i + h_i = 1, 2, \ldots, n \quad \text{(1)}$$
$$t_j - t_i - d_{ij} + h_j - (1 - X_{ij})M \text{ if } i, j = 1, 2, \ldots, n \text{ and } i \neq j \quad \text{(2)}$$
$$X_{ij} + X_{ji} = 1 \text{ if } i, j = 1, 2, \ldots, n \text{ and } i \neq j \quad \text{(3)}$$
$$X_{ij} \in \{0, 1\} \text{ if } i, j = 1, 2, \ldots, n \text{ and } i \neq j \quad \text{(4)}$$

Objective of Problem YCS is to minimize the sum of job waiting times. Constraints (1) give relationship between a job’s completion time, ready time and handling time. Constraints (2) give relationship between completion time of a job and that of its successors. Constraints (3) ensure correctness of precedence relationship specified by $X$. Constraints (4) is simple binary constraint. As the set of $t_i$, $i = 1, 2, \ldots, n$, that minimizes $\sum_{i=1}^{n} t_i$, total completion time would also minimize $\sum_{i=1}^{n} (t_i - h_i - r_i)$ and Problem YCS can be simplified as

$$\text{Minimize } \sum_{i=1}^{n} t_i \text{ subject to constraints (1) — (4)}.$$ 

Problem YCS is a problem of non-preemptive scheduling with different ready times on a single machine to minimize total completion time. Scheduling is an NP-complete problem\textsuperscript{14}.

**Novel Particle Swarm Optimization (NPSO) Algorithm**

PSO is an evolutionary algorithm, which is initialized with a population (swarm in PSO) of random solutions and searches for optima by updating generations. Each individual or potential solution, named particle, flies in dimensional problem space with a velocity dynamically adjusted according to the flying experiences of its own and its colleagues. PSO algorithm mimics the behavior of flying birds and their means of information exchange to solve optimization problems.

**Original Particle Swarm Optimization Algorithm**

Let the searching space is D-dimensional and $m$ particles form the colony. The $i$th particle represents a D dimensional vector $X_i$ ($i = 1, 2, \ldots, m$). It means that $i$th particle locates at $X_i = (x_{i1}, x_{i2}, \ldots, x_{id})$ ($i = 1, 2, \ldots, m$) in

---

**Fig. 1**—Typical container flows in terminal operations
searching space. The position of each particle is a potential result. Particle’s fitness is calculated by placing its position into a designated objective function. The ith particle’s flying velocity is also a D-dimensional vector, denoted as \( V_i = (v_{i1}, v_{i2}, \ldots, v_{iD}) \). Let the best position of ith particle is denoted as \( P_i = (p_{i1}, p_{i2}, \ldots, p_{iD}) \), and best position of the colony as \( P_g = (p_{g1}, p_{g2}, \ldots, p_{gD}) \). Original PSO algorithm is performed as

\[
\begin{align*}
    v_{id}(k+1) & = w v_{id}(k) + c_1 r_1 (p_{id}(k) - x_{id}(k)) + c_2 r_2 (p_{gd}(k) - x_{id}(k)) \quad \ldots (5) \\
    x_{id}(k+1) & = x_{id}(k) + v_{id}(k+1) \quad \ldots (6)
\end{align*}
\]

where \( k \) represents iterative number, \( c_1, c_2 \) are cognitive learning rate and social learning rate respectively. Usually, \( c_1 = c_2 = 2 \). \( r_1, r_2 \) are random numbers between 0 and 1.

Termination criterion for iterations is determined according to whether the max number of generation or a designated max value of fitness of \( P_g \) is reached. From Eqs (5) and (6), it is not applicable for discrete variable and hence not compatible for scheduling problems like YCS. So, a different method has to be adopted to fly particles in search space. To achieve this, crossover and mutation operators from Genetic algorithms are employed.

Proposed Particle Swarm Optimization with Genetic Algorithm Operators (PSOGAO)

PSOGAO algorithm for YCS to minimize completion time is set in a discrete space, in order to develop an effective problem mapping and solution generation mechanism for YCS problem. Crossover and mutation operators are used in NPSO instead of Eqs (5) and (6) of original PSO due to the discrete nature of the problem.

Crossover Operator

Crossover operation requires two candidate solutions (parents). A segment of the first parent is randomly chosen and this string is copied on to the new solution (offspring). Then remaining part of offspring is completed in the same order as the second parent but after scanning offspring to avoid duplication. The sequence of number that is not yet present in the offspring is added in the mentioned order and schedule is completed.

Mutation Operator

In this implementation, two positions in the parent string (schedule) are randomly selected and the jobs in those positions are swapped to get the offspring (new scheduling).

Many other variations of crossover and mutation operators are available\(^{15}\), but in this application, basic crossover and mutation operators are applied. For each candidate solution crossover operation is performed. The candidate will be the first parent and the best solution is the second parent. The best candidate solution is chosen stochastically for crossover; it can be either the personal best or the global best. Offspring then replaces first parent in the population that will be evaluated in the coming iteration. Mutation is restricted to the particle, which has attained the global best. This way, mutation probability is very low and crossover is performed extensively to explore search space and also exploit in the region where best solution is obtained.

Results and Discussion

To present the results of simulation runs for evaluating the schedule, for which total finish time will be minimum the least, the process is two fold. In first phase, mathematical model for integer programming is
developed to calculate the total completion time for any schedule. In second phase, total completion time is optimized. PSOGAO was applied to optimize the total finish time for the given port layout. Ready time (truck arrival time) \( r_i \), the time required by the yard crane to handle jobs (trucks) \( h_i \) and the time required for the yard crane to travel from the location of job \( i \) to the location of job \( j-D_{ij} \) that is used for the simulation runs (Table 1).

PSO performance is sensitive to choices for control parameter\(^{16,17} \). PSOGAO parameters chosen for simulation runs are as follows: cognitive learning rate, \( C_p = 2 \); social learning rate, \( C_g = 2 \); inertia factor, \( W = 1 \); number of swarm particles, \( N = 10 \); max number of iterations, 500. Number of swarm particles and the other parameters are decided empirically. Various simulation runs of the algorithm are carried out. It can be observed (Table 2) that results contained different schedules in each run, attributed to stochastic nature of the algorithm. Completion times for all schedules are very similar to each other. Slight change in schedule will affect completion time. Results obtained by PSOGAO are better fit than those obtained by heuristic method\(^1 \). More than one schedule is obtained with slightly different total completion time. Advantage of such alternative solution is that operator has prior knowledge about limitations of yard crane and shipments that are arriving and can choose the best solution based on all these factors and not entirely based on completion time. Operator has flexibility to choose between schedules and not forced to operate with a given schedule as in heuristic method.

### Conclusions

A variant of particle swarm optimization with genetic algorithm operators (PSOGAO) has been presented to solve scheduling problem of a yard crane to handle jobs with different ready times within its movement zone. Computational results have shown that PSOGAO is an effective method to find better-fit schedules for YCS in comparison with conventional heuristic method.

#### Table 1—Simulation runs of yard crane scheduling

<table>
<thead>
<tr>
<th>Job location, ( j )</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>( r_i )</td>
<td>2</td>
<td>5</td>
<td>7</td>
<td>13</td>
<td>15</td>
</tr>
<tr>
<td>( h_i )</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>( D_{ij} )</td>
<td>2</td>
<td>0</td>
<td>3</td>
<td>4</td>
<td>7</td>
</tr>
<tr>
<td>( h_j )</td>
<td>1</td>
<td>1</td>
<td>4</td>
<td>1</td>
<td>4</td>
</tr>
<tr>
<td>( r_j )</td>
<td>2</td>
<td>4</td>
<td>7</td>
<td>4</td>
<td>3</td>
</tr>
</tbody>
</table>

\( r_i \), truck arrival time; \( h_i \), handling time; and \( D_{ij} \), time required to travel for yard crane

#### Table 2—Optimal schedules obtained for PSOGAO in various runs

<table>
<thead>
<tr>
<th>Sl No</th>
<th>Schedule obtained from PSOGAO</th>
<th>Completion time</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>4 1 5 2 3</td>
<td>83</td>
</tr>
<tr>
<td>2</td>
<td>5 1 4 2 3</td>
<td>83</td>
</tr>
<tr>
<td>3</td>
<td>1 4 2 3 5</td>
<td>84</td>
</tr>
<tr>
<td>4</td>
<td>4 1 5 3 2</td>
<td>89</td>
</tr>
<tr>
<td>5</td>
<td>4 1 3 2 5</td>
<td>79</td>
</tr>
<tr>
<td>6</td>
<td>3 2 5 1 4</td>
<td>87</td>
</tr>
<tr>
<td>7</td>
<td>4 2 5 1 3</td>
<td>87</td>
</tr>
<tr>
<td>8</td>
<td>4 2 3 5 1</td>
<td>85</td>
</tr>
<tr>
<td>9</td>
<td>3 2 4 5 1</td>
<td>83</td>
</tr>
<tr>
<td>10</td>
<td>4 2 3 1 5</td>
<td>75</td>
</tr>
</tbody>
</table>
References