Automation in Construction 47 (2014) 69–77

Contents lists available at ScienceDirect

Automation in Construction

journal homepage: www.elsevier.com/locate/autcon

Enhanced crane operations in construction using service request optimization

Amir Zavichi [a,*], Kaveh Madani [b], Petros Xanthopoulos [c], Amr A. Oloufa [a]

[a] Department of Civil, Environmental and Construction Engineering, University of Central Florida, Orlando, FL 32816, USA
[b] Centre for Environmental Policy, Imperial College London, London SW7 2AZ, UK
[c] Department of Industrial Engineering and Management Systems, University of Central Florida, Orlando, FL 32816, USA

ARTICLE INFO

Article history:
Received 20 September 2013
Received in revised form 13 May 2014
Accepted 25 July 2014
Available online 24 August 2014

Keywords:
Tower cranes
Decision support system
Optimization
Traveling Salesman Problem
Optimization

ABSTRACT

This paper develops a novel service request sequence optimization model for tower crane operation efficiency improvement. The suggested model uses integer programming and modifies the classical Traveling Salesman Problem (TSP) formulation for optimizing construction tower crane operations. Numerical examples demonstrate that the application of the developed optimization model can result in 25–45% saving in the total travel time of the tower crane depending on the number of simultaneous requests.

1. Introduction

Today, with the necessity of timely, on-budget and high quality operations in construction projects, effective use of construction equipment is essential to successful completion of projects. Construction equipment alone places a great financial burden on projects and can cause economic losses if not utilized efficiently. Cranes are among the costly construction equipment, playing an important role in construction sites, especially in high-rise building projects. Activities that depend on cranes are usually on the project's critical path. Thus, improving crane operations can enhance project performance significantly.

Construction cranes are classified into tower and mobile cranes. Tower cranes are popular due to their high horizontal and vertical reachability, as well as their small footprint, especially in dense areas around the world [15]. The crane operation cycle consists of two work modes: stationary and dynamic. The stationary mode is experienced during loading or unloading when the hook does not have any motion while the dynamic mode is experienced when the hook is moving, including hoisting (vertical), trolleying (radial) and slewing (circular) movements. The total time associated with the crane's dynamic mode comprises the crane's travel time in a working cycle.

In this paper, we investigate the impact of prioritizing the crane service sequence on the overall crane's travel time using an innovative optimization method, which specifically modifies the Traveling Salesman Problem (TSP) optimization model, for maximizing the efficiency of construction tower crane operations. For this purpose, an exact combinatorial optimization method is proposed to overcome the existing constraints of the general TSP model that make it inapplicable to construction crane service sequencing. This model can assist on-site managers and crane operators in reducing the crane travel time through crane service sequence optimization. Reducing the crane travel time yields a shorter crane cycle and causes shorter delays in material delivery to downstream crews, increasing the total productivity of crane operations as well as those activities in need of crane services [16].

2. Background

Automating, planning and scheduling crane operations in order to improve total operation efficiency is of major interest due to the fact that cranes are the most instrumental material handling and lifting equipment in construction projects. Their importance is not only due to their high cost but also due to the central role they play in transporting material on project sites. Previous research on crane operation improvement falls into two main categories: crane layout pattern optimization and physical crane motion planning [21].

Crane layout pattern optimization deals with finding the optimum crane location out of the available alternatives in order to satisfy criteria such as balancing the workload and reducing the total crane operation time, or minimizing the spatial conflicts between cranes and other moving resources on the site. Zhang et al. [22] used a Monte-Carlo

* Corresponding author.
E-mail addresses: amir.zavichi@knights.ucf.edu (A. Zavichi), k.madani@imperial.ac.uk (K. Madani), petrosx@ucf.edu (P. Xanthopoulos), amr.oloufa@ucf.edu (A.A. Oloufa).

http://dx.doi.org/10.1016/j.autcon.2014.07.011
0926-5805/

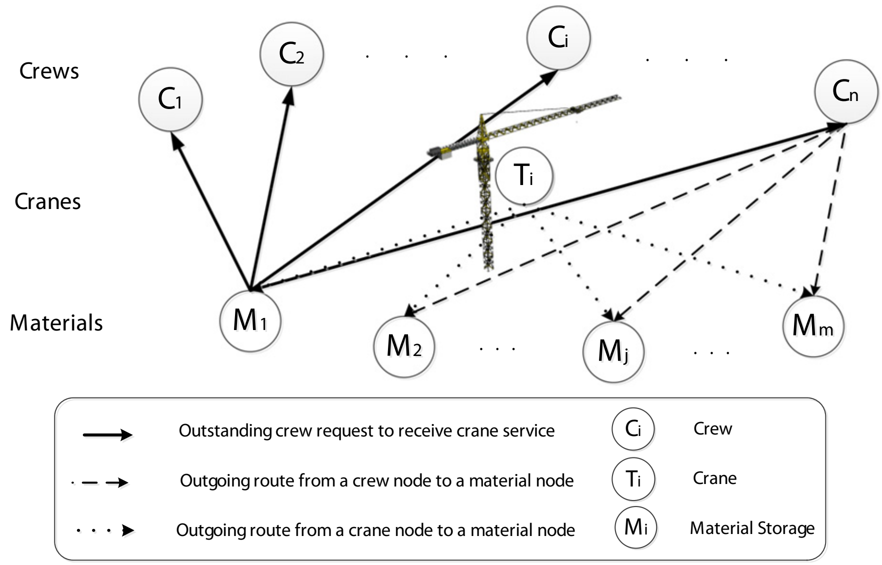

Fig. 1. Graphical illustration of a construction site layout.

simulation model to optimize the location of a single tower crane. In another study, Zhang et al. [23] performed a location optimization analysis for a group of tower cranes. Tam et al. [19], Tam and Tong [18], and recently Huang et al. [4] used various techniques to optimize the locations of a single tower crane and several supply points, keeping the demand locations fixed.

Physical crane motion planning aims to develop methods and tools that help the crane operator navigate the crane motions from the load pick-up to delivery. The technologies and methods that have been used to improve physical crane motion planning can be categorized into three categories: vision enhancement to provide a better view of the job site to the crane operator and reduce the need for a signal person [2,16]; automatic or semi-automatic navigation to ensure smooth maneuvering between loading and unloading locations [13,14]; and motion planning and collision avoidance to provide a path between loading and unloading locations while avoiding collision with objects surrounding the crane [6,9,11,17].

Another potential way to improve crane operation efficiency is minimizing the distance and time that the crane travels through appropriate ordering of the sequence of locations which the crane hook must visit in order to fulfill the service requested by the crews on a job site. A brute-force/exhaustive search method with the capability of analyzing up to 15 simultaneous requests was proposed by Zavichi and Behzadan [21]. This method addressed the problem of determining the sequence of items to be relocated from their existing locations to their newly assigned locations using a tower crane, such that the total travel time is minimized. With the same objective, this paper presents a general mathematical model based on the well-known Traveling Salesman Problem (TSP) method for sequencing crane service requests with the ability of considering a relatively large number of requests in order to minimize the total crane travel time leading to the reduction of the total idle time of the on-site crew and equipment.

3. Crane service sequencing problem (CSSP)

The low efficiency of crane operations has inverse impacts on the construction project time and budget. Crane operation efficiency is influenced not only by the crane operator's skills in navigating the crane but also by the decisions that the operator makes during operations.

Current construction crane operations are somewhat chaotic in many parts of the world. In most cases, there is no strong preplanning and scheduling of crane operations and service sequencing are conducted in real-time. In case of equally important tasks, the operator might make the decision on sequencing the requests based on his/her instincts or using heuristic scheduling rules, e.g., the FIFO (first-in-first-out) rule. In other cases, when some level of preplanning exists, requests are sent to the superintendent in advance and crane operations are scheduled through coordination between the crane operator and the superintendent who tries to maximize the operation efficiency. New requests during the operations are addressed on an ad hoc basis, depending on their priorities and crane availability. This scheduling process is significantly inefficient and normally far from optimal. Therefore, the need for a planning method to enhance the crane service scheduling is eminent. The recent technological and computational advancements enable us to

Crew	Material Request
1	2
2	3
3	1

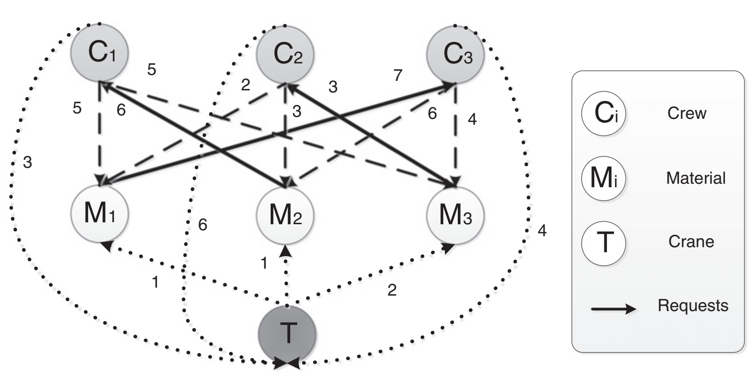

Fig. 2. Travel time graph and service request matrix for the example CSSP.

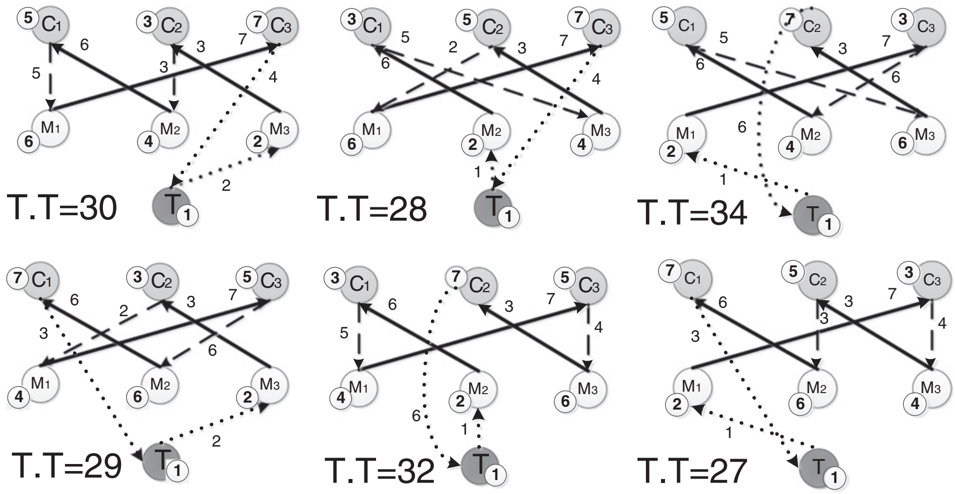

Fig. 3. Possible request fulfillment sequences and their total travel time (T·T) for the CSSP example.

address this need effectively. For example, the available smartphone technologies allow for real-time submission and update of the requests by the crew, allowing for continuous updating of the crane operation schedule using optimization models. These models can be efficiently solved with the existing computational capacity that let us solve many unsolvable optimization problems of the past in only a few seconds [12]. Thus, this work tries to develop a new framework for real-time and efficient management of crane operations using optimization models.

The crane service sequence problem can be best described using a graph, i.e. collection of vertices and connecting edges, associated with travel times. Fig. 1 shows a site layout using a graph consisting of crew (C) and material (M) nodes in a construction site in which material is delivered from *m* storage points to *n* working crews based on their requests using a central crane. Each crew node sends its requests to the crane operator to receive certain materials. The operator should then decide the order of request fulfillment, trying to minimize the total operation time as an objective with respect to different constraints such as the due time and precedence of tasks.

Assuming that a subset of crews (*w* out of *n*) request material delivery, there are *w* alternatives for the decision maker (crane operator) to pick from as the first delivery service point. Once the first target crew node is chosen, the operator must load the requested material by this crew and deliver it to the target node. If there is no new request, the process continues with *w-1* requests until all the outstanding crane service requests are fulfilled. Given that the operator is free to choose any order of deliveries, there is a total of *w*! (permutation of *w*) possible ways to fulfill all requests. In order to minimize the overall travel time, the operator needs to find the optimal delivery sequence, which is not possible without serious computations even in small problems. For example, consider the small crane service sequencing problem (CSSP) depicted in Fig. 2. This problem includes one crane, three request (crew) nodes, and three material loading nodes. The travel time associated with the crane movement between each two nodes are shown next to the arcs connecting the nodes. Solid lines represent the outstanding crew requests (e.g., crew 1 needs material from material storage 2 and crew 2 needs material from material storage 3), dashed lines indicate the crane travel routes from the crew nodes to the material nodes, and dotted lines are outgoing/incoming routes from/to the crane nodes. The outgoing routes from the tower crane node relate to the crane movement at the beginning of each service batch or at the beginning of the workday. The incoming routes to the crane node (from crew nodes to the crane node), on the other hand, relate to the crane movement at the end of each service batch at the end of the workday or when there is no outstanding request in the system.

In this problem, there are 3! possible movement sequences to fulfill the requests. Fig. 3 shows sample sequences, each with a different total travel time. The crane operator must decide the order of locations to be visited in order to fulfill all outstanding requests while minimizing the total travel time. This problem only has one optimal solution with a total travel time of 27 time units. For this small problem, the optimal order can be found through enumeration. However, the possible sequences grow significantly with increase in the number of requests. Thus, finding the optimum delivery order with the least travel time through enumeration (brute-force search) as done in previous research [21] is not mathematically efficient.

In practice, different heuristics might be used in order to obtain the near-optimal operation sequence in the absence of crane operation optimization tools. Three heuristic rationales for ordering the deliveries include fulfilling requests based on the first-in first-out/served (FIFO) method, fulfilling the nearest neighbor's request next or the nearest neighbor first (NNF) method, and fulfilling the request with the shortest travel time next or the shortest job first (SJF) method. While these methods can improve the crane operation efficiency to some extent,

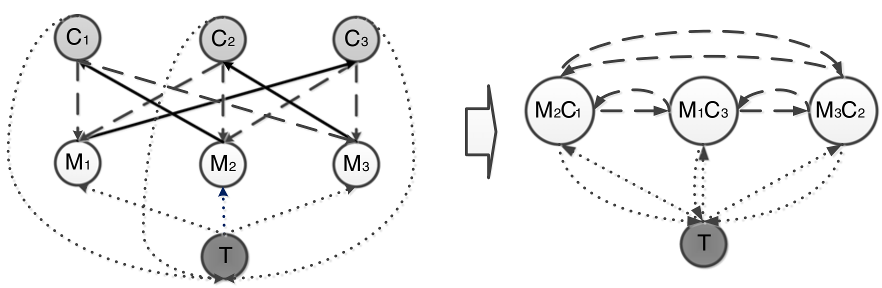

Fig. 4. Transforming a sample CSSP (left) to a TSP (right).

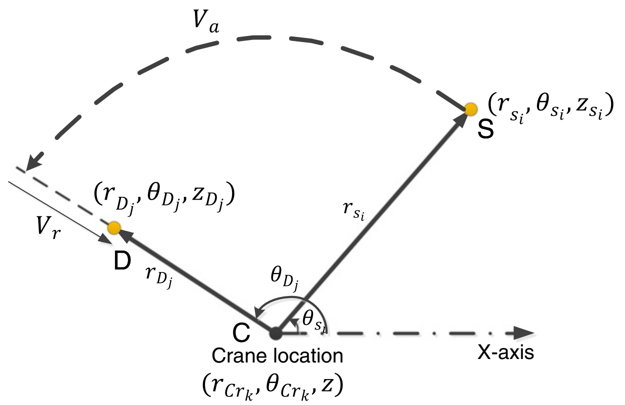

Fig. 5. Polar crane coordinates.

they do not generally result in the shortest path and least completion time, and do not guarantee an optimal solution [3]. Thus, the main objective of this paper is to develop an optimization method for solving the CSSP. To examine the efficiency of the heuristic order sequencing methods, crane travel time based on these methods is compared to the optimal travel time calculated using the optimization model proposed in this study. This will also help identify the best heuristic method for order sequencing in the absence of optimization models.

4. Method

In principle, the CSSP is similar to the Traveling Salesman Problem (TSP)—a well-known combinatorial optimization problem. In TSP, the salesman starts from an initial location, visits a prescribed set of cities, and returns to the original location in such a way that the total distance traveled is minimized and each city is visited only once [3]. TSP is one of the most notorious problems in Operation Research for being easy to explain but hard to solve [20]. This problem is an NP-complete problem that has become representative of difficult combinatorial optimization problems. In TSP, starting at city 1, the salesman has n − 1 choices for the second city to visit and n − 2 choices for the third city to visit, and so on. Thus, there are (n − 1)! possible tours in case of asymmetry (when the distance from city i to j is not equal to the distance from city j to i) and (n − 1)!/2 possible tours in case of symmetry.

CSSP can be formulated as a TSP, assuming that each request (starting from a material node and ending in a crew node) is a city and travel time associated with each link is the distance between two cities connected by the link. Based on this approach, the crane hook should fulfill requests by visiting request nodes and return to its initial position once no other outstanding request is left. This converts the CSSP to an asymmetric TSP, in which the cost (travel time) of moving from i to j is different from the cost of moving from j to i.

Fig. 4 shows how a sample CCSP can be converted to a TSP based on the suggested approach. The following points must be noted about the resulting TSP:

- Unlike TSP, cost (distance) is not only associated with the edges, but also with the nodes in the resulting TSP. The time associated with an edge relates to the travel time from a delivery node to a new request node while the time associated with each node represents the travel time from a material node to the target delivery node requesting that material. The optimization problem targets minimizing the total time of traveling on the edges while the node-specific times are constant and not sensitive to routing options.
- Unlike TSP, in CCSP the crane (corresponding to the salesman in TSP) can visit one node (crew or material) more than once.
- CCSP is inherently asymmetric regardless of the initial graph characteristics.

The first step in solving CSSP is to develop a travel time (cost or distance) matrix (C) associated with the connecting arcs in the original CSSP problem (prior to conversion). This matrix is referred to as the location travel time matrix, which is a square ($n \times n$) matrix in the following form:

$$C = \begin{bmatrix} c_{11} & \cdots & c_{1n} \\ \vdots & c_{\text{ij}} & \vdots \\ c_{n1} & \cdots & c_{nn} \end{bmatrix}$$

where n is the number of nodes in the graph and c_{ij} reflects the time that it takes for the crane hook to travel from node i to node j.

4.1. Hook travel time calculation

Reducing the crane operation cost involves minimizing the transportation time of the crane for which a reliable estimation of the hook travel time is required. Statistical and analytical models [10] have been applied for crane travel time estimation. In statistical models, the main driving variables (e.g. loading and unloading locations, crane trolley velocities (vertical, angular and radial), site conditions, and operator's skill level) are identified based on knowledge from field studies and a regression model is used to examine the correlations between these variables and travel time. Leung and Tam [10] used multiple linear regression models and Tam et al. [19] developed a nonlinear neural network model to predict the relationship between the driving factors, as independent variables, and transportation time, as a dependent variable. In analytical models, the number of variables is limited compared to the statistical models. Zhang et al. [21] developed an analytical model for tower cranes using the Cartesian coordinates of the supply, demand and crane locations. Since 1996, this mathematical model has remained almost intact and has been used in different studies [4,18,19,23]. Similarly, a polar coordinate system is used in this study in order to build the location travel time matrix.

4.1.1. Modeling transportation time using a polar coordinate system

Fig. 5 shows a polar coordinate system with pole C and polar axis X, where C is the crane's base location $(r_{Cr_k}, \theta_{Cr_k}, z)$ and X is an arbitrary fixed direction from which other angles are measured. Cartesian locations (x, y, z) are used to calculate the radial distance (r) and angle $(\underline{\theta})$

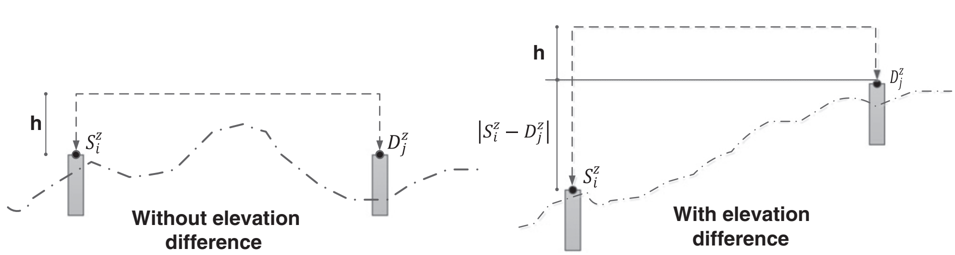

Fig. 6. Hoisting height for supply and demand without/with elevation difference.

using Eqs. (1) and (2):

$$r = \sqrt[2]{x^2 + y^2} \tag{1}$$

$$\theta = \tan^{-1}\left(\frac{y}{x}\right) \tag{2}$$

where $\tan^{-1}\left(\frac{y}{x}\right)$ is interpreted as the two-argument inverse tangent which takes into account signs of *x* and *y* to determine the quadrant which *θ* lies in.

Velocities in each direction can be based on the crane's manufacturing specifications (radial V_r(m/min), angular V_a(rpm), and vertical V_v(m/min) velocities). Radial ($T_r^{(i,j)}$) and angular ($T_a^{(i,j)}$) components of the hook travel time between two locations i and j are calculated using Eqs. (3) and (4), respectively:

$$T_r^{(i,j)} = \frac{\left|r_{S_i} - r_{D_j}\right|}{V_r} \tag{3}$$

$$T_a^{(i,j)} = \frac{\left|\theta_{S_i} - \theta_{D_j}\right|}{V_a}. \tag{4}$$

To rectify the possible underestimation of the Zhang et al.'s model [22] in estimating the vertical component of the hook travel time, in problems with supply and demand node elevation differences (Fig. 6), an extra motion (minimum hoisting height) is added to both sides of the travel arcs (both loading and unloading locations). The minimum hoisting height depends on the type of material (e.g., loading steel bars requires a higher hoisting height than loading materials using a bucket), site topography, obstructions on the job site, and safety factors. Fig. 6 shows how the minimum hoisting height for the loading and unloading points can vary on job sites with and without elevation differences. Given that the minimum hoisting height is traversed two times, the vertical component of travel time can be calculated using Eq. (5):

$$T_v^{(i,j)} = \frac{\left(\left|S_i^z - D_j^z\right| + 2 \times h\right)}{V_v} \tag{5}$$

where *h* is the minimum hoisting height.

Three parameters are used to account for operator's skill (α and β) and the site conditions (γ). α is the degree of overlap in radial and angular movements, i.e., to what extent the operator can simultaneously move the hook in both radial and angular directions. The travel time in the horizontal plane can be calculated using Eq. (6):

$$T_h^{(i,j)} = \max\left\{T_r^{(i,j)}, T_a^{(i,j)}\right\} + \alpha \cdot \min\left\{T_r^{(i,j)}, T_a^{(i,j)}\right\}. \tag{6}$$

Parameter β is used to take into account the operator's skill in simultaneous movement of the hook in horizontal and vertical planes. Travel time can change based on the working site conditions such as weather conditions, existence of obstacles and various safety issues. Parameter γ is used to account for working site conditions [4]. Total travel time, which is the combination of the horizontal and vertical movement times, can be calculated using Eq. (7):

$$T_{(i,j)} = \gamma \cdot \left(\max\left\{T_h^{(i,j)}, T_v^{(i,j)}\right\} + \beta \cdot \min\left\{T_h^{(i,j)}, T_v^{(i,j)}\right\}\right) \tag{7}$$

where $0 \le \alpha \le 1$, $0 \le \beta \le 1$, and $1 \le \gamma \le \infty$.

α, β, and γ are continuous positive numbers. Lower values of α and β reflect a higher degree of simultaneity in movements in two directions. Lower values for parameter γ reflect more convenient operation conditions, e.g. the value of 1 is used for normal weather conditions in an open area without on-site obstructions. The values of α and β and γ need to be estimated based on observations in the construction job site. Using Eqs. (1)–(7) the location travel time matrix (C: (c_{ij})) can be developed.

4.2. CSSP formulation

The CSSP can be represented by a directed graph $G = (V, A)$, where V is the set of n vertices (representing requests) and A is the directed arc set. The mathematical formulation of the optimization model to solve the CSSP is as follows:

$$\text{Minimize} \qquad \sum_k \sum_l p_{kl} y_{kl} \tag{8}$$

Subject to

$$\sum_{l:l \ne k} y_{kl} = 1 \qquad \forall k,\ l \in V, (k,\ l) \in A \tag{9}$$

$$\sum_{k:k \ne l} y_{kl} = 1 \qquad \forall k,\ l \in V, (k,\ l) \in A \tag{10}$$

$$\sum_{k \in S} \sum_{l \in \bar{S}} y_{kl} \ge 1 \quad S \subset V,\ 2 \le |s| \le n-2,\ (k,\ l) \in A \tag{11}$$

$$y_{kl} \in (0,\ 1) \qquad \forall k,\ l \in V,\ k \ne l \tag{12}$$

where P:p_{kl} is the service request matrix associated with A.

The suggested integer optimization model determines the minimum cost (travel time) circuit that fulfills each request once and only once. Such a circuit is known as a *tour* or *Hamiltonian circuit* (or *cycle*) [7]. Integer programming is a well-known method in the classic optimization literature. The entire class of problems referred to sequencing, scheduling, and routing are inherently integer programs. In this problem, a binary decision variable y_{kl} is associated with every arc (k,l), and is set equal to 1 if and only if arc (k,l) is used in the optimal solution ($k \ne l$). In other words, $y_{kl} = 1$ if the crane hook goes directly from request node k to request node l, and $y_{kl} = 0$ otherwise (constraint 12). Constraints 9–10 are degree constraints which specify that every vertex is incident of one outgoing arc (constraint 9) and one ingoing arc (constraint 10). The solution considering only constraints 9 and 10 might lead to a disconnected solution (subtour) that needs to be excluded from the set of solutions. To eliminate the solutions that consist of subtours (i.e., tours on subsets of less than n vertices), an additional constraint (constraint 11) is needed. S is a subset of V vertices and |s| is the cardinality of S. In addition, $\bar{S}$ is a complement of S. Constraint 11 is only valid when $2 \le |s| \le n-2$ and prevents the solution to contain two or more disjoint subtours.

The CSSP travel time matrix (service requests matrix (P:p_{kl})) is a dynamic matrix composed of the location travel time matrix (C:(c_{ij})) combined with the requests at a given time. The former matrix reflects the travel time between request nodes. This matrix is inherently asymmetric ($P^T \ne P$).

As mentioned before, CSSP must be converted to an asymmetric TSP here. However, asymmetric TSPs are complex to solve. One way of solving an asymmetric TSP is to double the size of the distance matrix by replacing every node in the graph with two nodes [5], having the added nodes represent dummy cities. The links between each node and its corresponding duplicated dummy node is associated with super low travel costs (− ∞). This assures that a real node and its corresponding dummy node are passed through after each other in the final sequence. The original distances given in the service request matrix (P:p_{kl}) are used for distances between the nodes and the duplicated dummy nodes, where paths start from real nodes and end in the duplicated dummy nodes. The distances between real nodes and between all

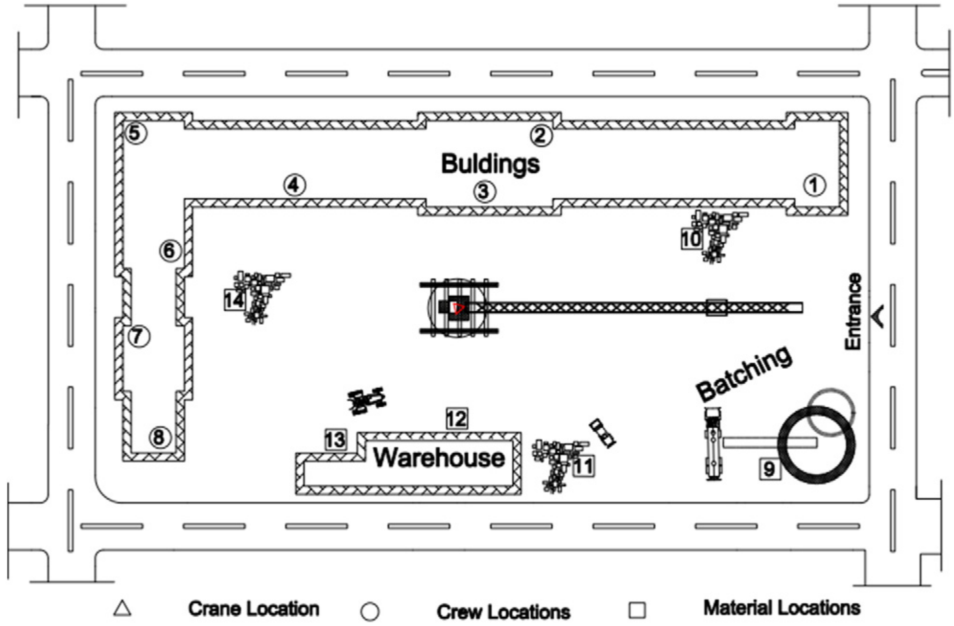

Fig. 7. Site layout.

duplicated nodes are assumed to have very large costs ($+\infty$) since there is no path between them. This procedure transforms the asymmetric matrix to a symmetric one. As an example, the following asymmetric matrix for a TSP with four nodes (left) can be converted to a symmetric matrix (right) through the explained procedure.

$$\begin{pmatrix} 0 & d_{12} & d_{13} & d_{14} \\ d_{21} & 0 & d_{23} & d_{24} \\ d_{31} & d_{32} & 0 & d_{34} \\ d_{41} & d_{42} & d_{43} & 0 \end{pmatrix} \Leftrightarrow \begin{pmatrix} & & & & -\infty & d_{21} & d_{31} & d_{41} \\ & & +\infty & & d_{12} & -\infty & d_{32} & d_{42} \\ & & & & d_{13} & d_{23} & -\infty & d_{43} \\ & & & & d_{14} & d_{24} & d_{34} & -\infty \\ -\infty & d_{12} & d_{13} & d_{14} & & & & \\ d_{21} & -\infty & d_{23} & d_{24} & & & +\infty & \\ d_{31} & d_{32} & -\infty & d_{34} & & & & \\ d_{41} & d_{42} & d_{43} & -\infty & & & & \end{pmatrix}.$$

Once an asymmetric TSP is converted to a symmetric one, the minimum transportation cost and its associated travel route can be found using the proposed optimization model.

5. Numerical evaluation

To underline the utility of the suggested optimization model, its performance is compared to three conventional heuristic scheduling methods, namely the FIFO, SJF, and nearest neighbor NNF that are prevailing approaches for scheduling problems. The FIFO algorithm involves no intelligence and the crane operator processes the requests based on the received order. Based on the SJF algorithm, the request with the smallest estimated travel time has the highest priority to be fulfilled. Based on the NNF algorithm, crane operator should serve the next closest request after each delivery.

5.1. Computational experiment: a predefined facility layout case

A numerical example with six material (supply) locations and eight demand (crew) locations is considered here. In this problem (Fig. 7) one central tower crane is in charge of transporting the materials. For simplicity, the initial tower crane hook location is assumed to be at (0,0,0) and other locations are determined with respect to that. Table 1 provides the material and crew coordinates in this problem. Specifications of the heavy-load 4000 HC 100 Liebherr tower crane are used for crane velocities: $V_v = 136$ m/min, $V_a = 0.5$ revolution/min and $V_r = 60$ m/min. α and β are assumed to be 0.25 and 1, respectively, based on previous studies [4,19,23]. γ is set equal to 1, assuming the site conditions are normal [4]. Using the transportation model described in Section 4.1.1 the hook travel time between the nodes can be calculated to develop the location travel time matrix. The minimum hoisting height is considered to be 5 m for this experiment ($h = 5$ m).

To evaluate the performance and utility of the proposed optimization model for solving CSSP with different sizes, the numerical example is solved for 11 different sizes with different number of requests (i.e., 10, 20, 30, 40, 50, 100, 200, 300, 400, 500, and 1000 requests). Using a uniform probability distribution, random requests are generated for each CSSP with a given size. To ensure that the generated problems are random, 100 CSSPs are generated and solved for each problem size, making the total number of solved problems 1100 (100×11). Each of the 1100 CSSPs are then solved using the suggested optimization model as well as the three heuristic crane operation algorithms. The mean and standard deviations of the total travel times are determined for different problem sizes using all the scheduling methods, i.e. FIFO, SJF, NNF, and CSSP optimization. The optimization model is solved using CONCORDE, a symmetric TSP exact solver [1] in integration with MATLAB.

Average savings in total travel time for different number of requests under different scheduling methods with respect to the travel time under FIFO method are presented in Table 2. To facilitate the comparison, the total travel time based on the FIFO scheduling method is set as the baseline and other scheduling methods are compared to this baseline. Results show that the intelligence added to the sequence processing reduces the average travel time by 9%, 32%, and 35% using the SJF, NNF, and the optimal scheduling methods in comparison to the FIFO method, respectively, for different numbers of requests in. As can

Table 1
Coordinates of the supply and demand locations.

Material location (Supply)	Location (x,y,z)	Crew location (Demand)	Location (x,y,z)
1	(76,−39,0)	1	(86,29,10)
2	(57,16,0)	2	(20,41,5)
3	(30,−39,0)	3	(6,29,3)
4	(0,−27,0)	4	(−40,30,12)
5	(−30,−32,0)	5	(−79,42,4)
6	(−55,1,0)	6	(−70,13,5)
		7	(−77,−7,0)
		8	(−72,−32,0)

Table 2
Average time saving under different scheduling methods (SJF, NNF, and optimization) for the pre-defined site layout example.

Number of requests	Time saving relative to the FIFO method (%)			Run time (sec.)
	SJF	NNF	Optimization	
10	−1%	20%	24%	0.16 ± 0.1
20	3%	24%	28%	0.21 ± 0.33
30	3%	26%	30%	0.27 ± 0.45
40	4%	29%	33%	0.56 ± 1.35
50	5%	30%	34%	0.83 ± 1.6
100	8%	35%	37%	6.42 ± 12.73
200	12%	37%	38%	11.14 ± 14.7
300	14%	38%	40%	29.71 ± 43.17
400	16%	39%	40%	105.2 ± 147.2
500	17%	40%	41%	258.4 ± 195.6
1000	19%	36%	41%	1727.48 ± 1129.6

be seen, the proposed optimal scheduling method outperforms the other methods. The NNF heuristic rule also provides a satisfactory sequence scheduling without a need for optimization. The standard deviations for all problem sizes were less than 7% of the mean travel time. In addition, the standard deviation decreased as the number of requests increased.

Table 2 also reports the mean computational time and its standard deviation for each problem size. Small computation times show the applicability of the suggested crane service sequence optimization method in practice. Run times increased exponentially as the number of requests increased. The high variation in run times is due to the problem structure variability, mainly due to the inclusion of the subtour elimination constraint in the model.

To evaluate the significance of the optimized results in comparison to the FIFO approach, the *t*-test was performed and the significance level (p-value) was calculated. The significance level for all problem sizes was less than 10^{-15}. This suggests that the results are significantly different, rejecting the null hypothesis ($\mu_{FIFO} = \mu_{Optimal}$) for all problem sizes, which shows the consistency of the method in finding the optimal travel time.

5.2. Computational experiment: a random site layout

To show the independence of the optimization model performance from the site layout, the predefined (deterministic) site layout of Section 5.1 is replaced with a random site layout. To test the performance of the sequencing methods in case of random site layout, for each problem, random nodes, scattered around a central tower crane with a 70-m operation radius, are generated using a uniform probability distribution (Fig. 8). Each random site layout is assumed to have 50 nodes with random coordinates and elevation differences from 0 to 10 m with respect to the hook's initial idle position. Once random nodes are generated for a CSSP, random request pairs (i, j) are generated using a uniform distribution. Each request (i, j) pairs a supply node (i) with a demand node (j), where both i and j belong to the set of 50 generated random nodes. In this case, each node in the random site can serve both as a supply or demand location, depending on the randomly generated request pair. Similar to the previous case, the problem is solved 100 times for 11 different sizes. Each of the 1100 problems has a unique randomly generated layout with 50 nodes. Values of the other variables (crane velocities, operator skill parameter, etc.) are the same as the problem illustrated in the previous section.

Table 3 presents the results of the analysis for the random site layout problem. Similar to the previous example with predefined site layout, the standard deviations of time savings are within an acceptable range (10% in this case). Performance of the SJF method is not significantly better than the FIFO method, making it an inferior scheduling method compared to NNF and optimal scheduling. On average, optimization results in a time saving of 36%. This value is 33% for the NNF method as the

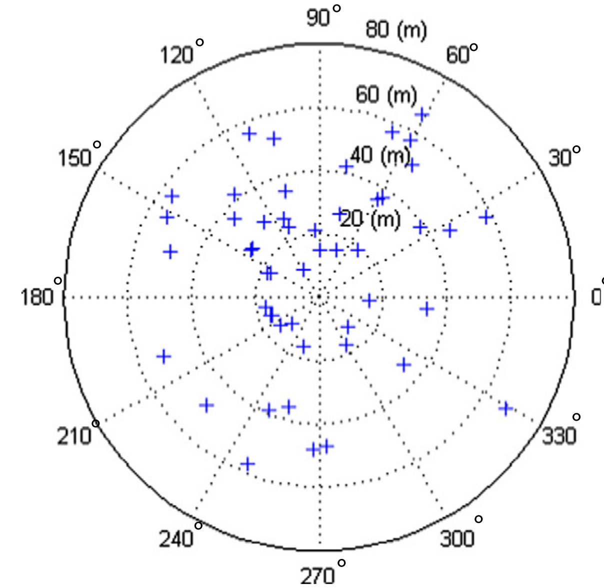

Fig. 8. Random node coordinates around a central tower crane.

best heuristic scheduling method. The time savings increase for all methods as the problem gets larger.

Here, the Concorde TSP solver was used for solving the CSSP problem. Concorde is considered as the best available solver for symmetric TSP [8]. The largest instance that has been successfully solved by Concorde had 85,900 vertices [8]. CSSP was originally an asymmetric TSP which was converted to a symmetric TSP here by doubling the size of the distance matrix. Therefore, logically we expect to be able to successfully solve CSSP problems with up to 42,950 requests using Concorde. The computational time was very small for small problems with up to 100 requests, making the model applicable in practice. In general, the obtained results for the random site layout are consistent with the results in the previous case, indicating the robustness of the performance of the suggested optimization model.

5.3. Sensitivity to input parameters

To further examine if the model output is sensitive to the input parameters (V_v, V_a, V_r, α, β, γ), a one-way sensitivity analysis is conducted. In the one-way sensitivity analysis, only one parameter changes at a time while other parameters remain constant and the impact of the change on the model output is examined. For simplicity, the problem size is fixed to 100 requests. The general modeling procedure is the same as the previous case with random site layouts. Each time, one input parameter of the model is changed by a given amount within a

Table 3
Average time saving under different scheduling methods (SJF, NNF, and optimization) for the random site layout example.

Number of requests	Solution algorithm			Run time (sec.)
	SJF saving (%)	NN saving (%)	Optimal saving (%)	
10	1%	18%	25%	0.71 ± 0.06
20	0%	24%	30%	0.82 ± 0.1
30	1%	27%	32%	0.93 ± 0.18
40	1%	29%	34%	1.16 ± 0.26
50	1%	30%	34%	1.9 ± 1.29
100	2%	35%	38%	8.01 ± 11.6
200	3%	38%	40%	45.33 ± 56.04
300	3%	39%	41%	128 ± 227.6
400	3%	40%	41%	293 ± 541.3
500	4%	40%	42%	346.9 ± 626.9
1000	8%	42%	43%	3269.4 ± 2267

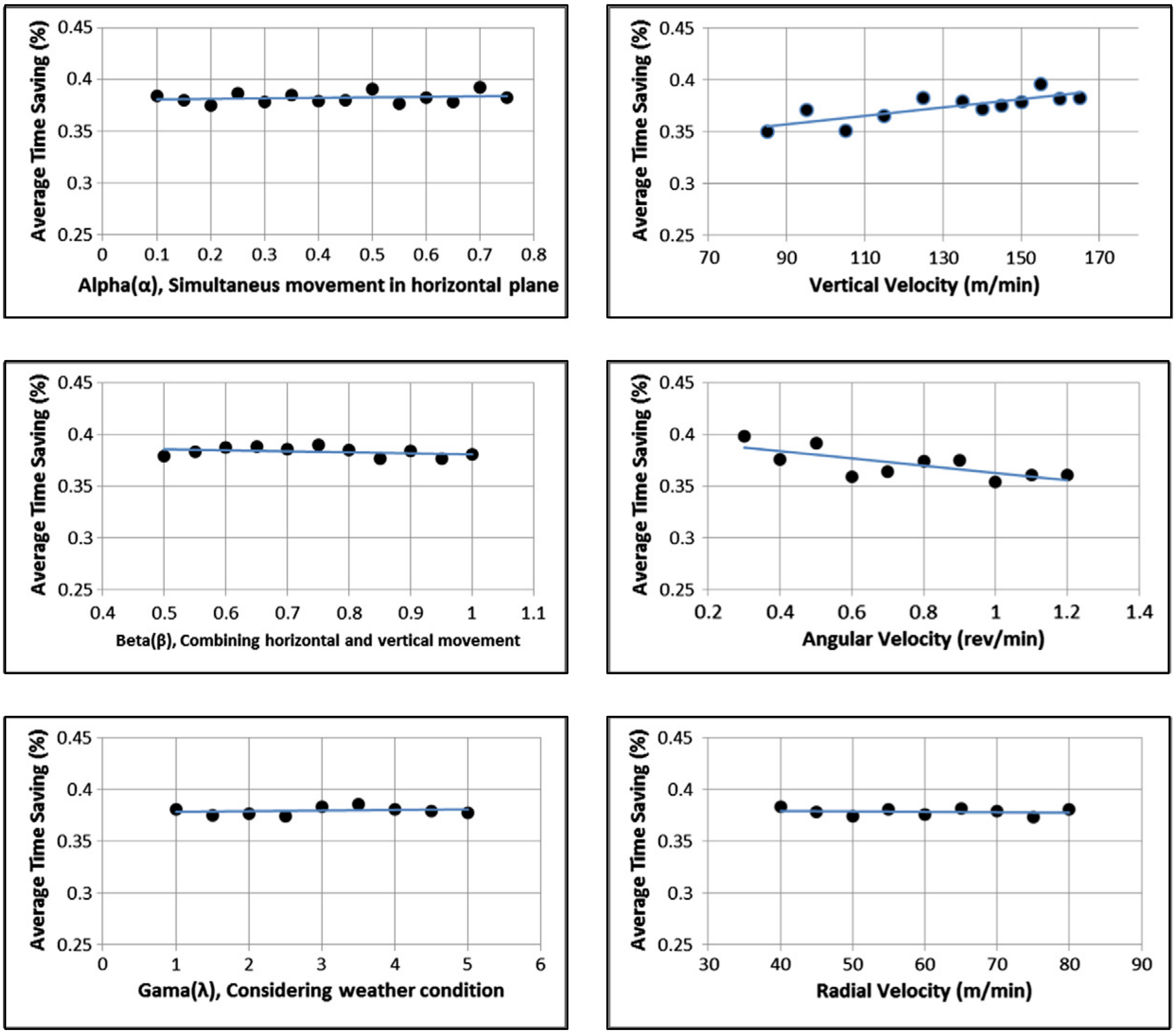

Fig. 9. Sensitivity analysis results.

meaningful range, and the mean value of time saving percentage for 100 trials is recorded. Fig. 9 shows the sensitivity analysis results. Each graph shows how the average time saving percentage with respect to the FIFO method varies by changing the value of one of the input parameters. While the model output is sensitive to the input parameters, the linear relationship between the input parameters and the time saving suggests that the model performance is not affected by input values and the output changes are consistent with the input changes. It must be noted that saving in operation time is expected through application of the proposed optimization model, independent of the crane model and its velocity specifications. Nevertheless, the level of time saving can change based on velocity specifications as the sensitivity analysis results indicate.

6. Conclusions

Use of an optimization technique to improve the crane operation efficiency via prioritizing job requests was proposed in this paper. An exact combinatorial optimization method, which is a modification of the "Traveling Salesman Problem (TSP)", was proposed for minimizing the = crane travel time by optimal ordering of the crane movement sequences.

The suggested optimization model results in 25–45% saving in the travel time in comparison with the conventional First-In-First-Out approach in fulfilling the requests. The model's performance is not highly sensitive to the input parameters and different jobsite specifications. The small run time of the optimization model makes it useful in practice, helping reduce crane operation time and crane-related activity costs considerably. The developed model optimizes the crane travel time only, which is a significant portion of crane cycle operations, especially in high rise constructions where the loading and unloading times constitute a small portion of the crane movement cycle.

While the proposed model was tested for the operations of a single crane, it can be extended to solve multi-crane operation scheduling problems. This would require solving an assignment problem as the primary step. The assignment problem in this case is the combinatorial optimization problem of finding the maximum weight matching in a weighted bipartite graph to reduce the total travel times considering some crane-specific constraints such as distance of the cranes to loading/unloading locations. The assignment problem is first solved to assign the requests to a specific crane and the assigned requests are then optimized using the service sequence optimization model developed in this study. In addition, the proposed model can be extended to consider other factors such as request deadlines, and dynamic or intermittent requests, which can be the subject of future studies.

Similar to any other modeling study, this study had some limitations and simplifying assumptions. Here, the travel time between two nodes was considered to be deterministic while the travel time can vary in practice. Future studies can consider stochastic travel times. Given that the time savings increase with an increased travel time resulting from elevation differences, future studies can investigate the effects of larger elevation differences (more than 10 m) on the travel time. This study assumed that each loaded bucket can be sent to one target location only, i.e., the crane hook does not visit multiple demand nodes after being loaded. Future studies might relax this assumption. To make the developed proof-of-concept model more practical, task deadline, sequence priority, and intermittent requests can be added to the problem formulation. While in this study the travel time was assumed to be independent of the load, future studies can evaluate the effects of material weight on the travel time. Finally, given the crane operation efficiency is strongly tied to the project duration and cost, future studies might consider evaluating this connection.

Acknowledgments

The authors would like to thank Aviad Shapira for his encouraging comments and helpful suggestions during this research. In addition, the authors would like to express their gratitude to William Cook for his support and help in using the TSP exact solver (CONCORDE). Constructive comments from the three anonymous reviewers are also appreciated.

References

[1] D. Applegate, R. Bixby, V. Chvatal, W. Cook, Concorde – a code for the solution of the TSP, Available at http://www.math.uwaterloo.ca/~bico/software.html (last accessed 08/14/2014).
[2] J.G. Everett, A.H. Slocum, Cranium — device for improving crane productivity and safety, J. Constr. Eng. Manag. Asce 119 (1) (1993) 23–39.
[3] G. Gutin, A.P. Punnen, The Travelling Salesman Problem and Its Variations, Springer, Newyork, 2004.
[4] C. Huang, C.K. Wong, C.M. Tam, Optimization of tower crane and material supply locations in a high-rise building site by mixed-integer linear programming, Autom. Constr. 20 (5) (2011) 571–580.
[5] R. Jonker, T. Volgenant, Transforming asymmetric into symmetric traveling salesman problems, Oper. Res. Lett. 2 (4) (1983) 161–163.
[6] S. Kang, E. Miranda, Planning and visualization for automated robotic crane erection processes in construction, Autom. Constr. 15 (4) (2006) 398–414.
[7] G. Laporte, The traveling salesman problem — an overview of exact and approximate algorithms, Eur. J. Oper. Res. 59 (2) (1992) 231–247.
[8] G. Laporte, A concise guide to the traveling salesman problem, J. Oper. Res. Soc. 61 (1) (2010) 35–40.
[9] Z. Lei, A Robotic Approach to the Analysis of Obstacle Avoidance in Crane Lift Path Planning Master Thesis, University of Alberta, Edmonton, Alberta, Canada, 2011.
[10] A.W.T. Leung, C.M. Tam, Prediction of hoisting time for tower cranes fro public housing construction in Hong Kong, Constr. Manag. Econ. (1999) 305–314.
[11] J. Olearczyk, Crane Lifting Operation Planning and Lifted Object Spatial Trajectory Analysis, (Doctoral Dissertation) University of Alberta, Edmonton, Alberta, Canada, 2010.
[12] D.E. Rosenberg, K. Madani, Water resources systems analysis: a bright past and a challenging but promising future, J. Water Resour. Plan. Manag. 140 (4) (2014) 407–409.
[13] Y. Rosenfeld, Automation of existing cranes: from concept to prototype, Autom. Constr. 4 (2) (1995) 125–138.
[14] Y. Rosenfeld, A. Shapira, Automation of existing tower cranes: economic and technological feasibility, Autom. Constr. 7 (4) (1998) 285–298.
[15] A. Shapira, G. Lucko, C.J. Schexnayder, Cranes for building construction projects, J. Constr. Eng. Manag. Asce 133 (9) (2007) 690–700.
[16] A. Shapira, Y. Rosenfeld, I. Mizrah, Vision system for tower cranes, J. Constr. Eng. Manag. Asce 134 (5) (2008) 320–332.
[17] P. Sivakumar, K. Varghese, N. Ramesh Babu, Automated path planning of cooperative crane lifts using heurisitc search, J. Comput. Civ. Eng. 17 (3) (2003) 197–207.
[18] C.M. Tam, T.K.L. Tong, GA-ANN model for optimizing the locations of tower crane and supply points for high-rise public housing construction, Constr. Manag. Econ. 21 (3) (2003) 257–266.
[19] C.M. Tam, T.K.L. Tong, W.K.W. Chan, Genetic algorithm for optimizing supply locations around tower crane, J. Constr. Eng. Manag. Asce 127 (4) (2001) 315–321.
[20] L.A. Wolsey, Integer Programming, John Wiley & Sons, New York, 1998.
[21] A. Zavichi, A.H. Behzadan, A real time decision support system for enhanced crane operations in construction and manufacturing, Proceedings of ASCE International Workshop on Computing in Civil Engineering, ASCE, Miami, FL, 2011, pp. 586–593.
[22] P. Zhang, F.C. Harris, P.O. Olomolaiye, A computer-based model for optimizing the location of a single tower crane, Build. Res. Inf. 24 (2) (1996) 113–123.
[23] P. Zhang, F.C. Harris, P.O. Olomolaiye, G.D. Holt, Location optimization for a group of tower cranes, J. Constr. Eng. Manag. 125 (2) (1999) 115–122.