KSCE Journal of Civil Engineering
Vol. 8, No. 4 / July 2004
pp. 365~369

Genetic Algorithm for Construction Space Management

By Hyoun-Seung Jang*

Abstract

Poor usage of space at construction sites is one of the leading causes of decreased productivity in the construction industry; however, it is manageable, and some instances preventable. This study focuses on how to efficiently manage space for construction materials on high-rise buildings in congested urban areas where space for materials around the building footprint is not available. The construction materials layout plan provides a logical order and priority for space planning decisions with reasonable costs. Genetic Algorithm (GA) modeling assumptions are made in order to properly allocate space for materials that will result in lower computational costs and increased in productivity.

Keywords: *construction materials, Genetic Algorithm (GA), layout plan*

1. Introduction

Construction productivity is greatly influenced by the organization of the project site and the flow of equipment, labor, and materials through the site. The most important resource is space since materials should be stored so they are accessible when needed. Materials storage area should be selected carefully to avoid impacting construction operations, and materials procured by the general contractor should be inspected upon delivery to ensure that the correct items and quantities were delivered. However, like any resource, the amount of construction site space demanded by the various activities changes with the schedule of the work. Therefore, as the schedule evolves during the project, the site layout may need to be efficiently re-organized at various intervals to satisfy the upcoming schedule requirements to maintain construction site productivity (Emad *et al*., 2001).

2. Problem Statement

Space and time conflicts have been identified as one of the major causes of productivity loss in construction (Ahuja and Nandakumar, 1984; Kuntz, 1994; Oglesby *et al*., 1989; Rad, 1980; and Sanders *et al*., 1989). Sanders *et al*. (1989) stated that efficiency losses of up to 65% are due to congested workspace and up to 58% are due to restricted access. Howell *et al.* (1993) reported the elimination of sharing resources, such as work areas, as a first step for performance improvement at construction sites, especially when a site is very tight or highly constrained such as construction in an urban environment or material rehabilitation. This paper presents a material layout model for multiple floor buildings in high-density urban areas.

3. Related Research

The most common study in space management in the industry is through site planning. These plans are necessary to manage not only space on the site for material deliveries, staging areas, and crane locations, but also space in high-rise buildings that are normally situated in congested urban locations. The results from literature over the last decade can be classified into two broad area of study: (1) Space-scheduling (Riley, 1994; Thabet and Beliveau, 1997) and (2) Site layout planning (Tommelein and Zouein, 1993; Tommelein *et al*., 1991; Yeh, 1995; Lin and Haas, 1996; Philip *et al*., 1997; Gero and Kazakov, 1997; Li and Love, 2000; Zouein and Tommelein, 2001; Tam *et al*., 2002; Zouein *et al*., 2002). However, this paper considers both space-scheduling and site layout planning which is shown in Fig. 1.

4. Multiple-Floor Material Layout Explore

In most cases, the approach is to take the shortest from one major activity to another. The objective of multiple-floor material layout is to minimize the total cost of material handling. The formulation of objectives can be expressed as (1):

$$Min\sum_{i=1}^{n}\sum_{j=1}^{m}W_{a_i f_j}D_{a_i f_j} \tag{1}$$

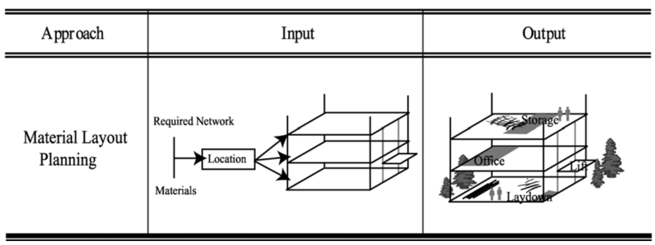


Fig. 1. Material Layout Planning Approach

Note: In construction, there are multiple meanings for the word material. For the purposes of this paper, a material is considered a direct or support resource to be used in construction. For example, a material could be a material pile, tool cart, or porta-potty.

*Member, Senior Researcher, Construction & Economy Research Institute of Korea, Korea (Corresponding Author, E-mail: jang@cerik.re.kr)

n: number of activities
m: number of facilities

$W_{a_i f_j}$ is the weight ordering between materials and activities. Coefficient $D_{a_i f_j}$ is the distance between an activity a_i and a material f_j. If a_i and f_j belong to the same floor, $D_{a_i f_j}$ is simply the horizontal distance between them. Otherwise, the sum of three distances is employed to $D_{a_i f_j}$: horizontal distance between f_j and the nearest lift, vertical distance between two floors, and the horizontal distance that lift is to a_i.

4.1. Assumptions

To improve the multiple-floor material layout model, assumptions are described below;

(1) A floor is divided into grids, so each material occupies at least one grid and there is no grid sharing.
(2) The size of materials is not equal and the area of activities is also not equal.
(3) The size of a material must be the smallest it can be but there is no limit to the number of materials of the same type. For example, a brick, which can be size 1, must be size 1 but there can be several brick materials each of them is size one.
(4) The populations of the materials whose sizes are bigger than one do not dominate the total population.
(5) There is no wall or obstacle so that the distance between two grids in the same floor is determined by a beeline.
(6) The distance between two materials whose sizes are bigger than one grid unit, is the average distance between the grid units they occupy.
(7) The shape of a material must be a square or rectangle.
(8) There is no limitation of number of lift but each lift operates entire floors. In another word, there is no lift such operate only between 3rd floor and 7th floor.
(9) If many lifts are located between floors, the lift which is on the shortest path is used.

4.2. Cost Function

The cost function for the multiple floor layout problems considers travel along the vertical direction as well as that in the horizontal direction. To transport material between materials located on different floors, a lift must be chosen from among the available vertical material handling devices. The cost function shown below uses the following notation:

$C_{a_i f_j}$ = The cost function
$D_{a_i f_j}$ = The distance between an activity a_i and a material f_j
$W_{a_i f_j}$ = The weight between an activity a_i and a material f_j
P_i = The three dimensional rectangular coordinate of f_i
P_l = The three dimensional rectangular coordinate of the closest lift
H, V = Horizontal and vertical cost of transporting a unit load through a unit distance
Z = The third dimension (X, Y, Z)

The total cost function $C_{a_i f_j} = \sum D_{a_i f_j} \cdot W_{a_i f_j}$, where a_i is an activity and f_j is a material.

$$C_{ij}\begin{cases} (z_i = z_j) & \|P_i - P_j\| \cdot W_{a_i f_j} \\ (z_i \neq z_j) & (\|(P_i - P_l)\cdot H\| + \|(P_j - P_l)\cdot H\| + \|(P_i - P_j)\cdot V\|)\cdot W_{a_i f_j} \end{cases} \quad (2)$$

$$H = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \end{bmatrix}, \quad V = \begin{bmatrix} 0 \\ 0 \\ k \end{bmatrix}$$

k = cost factor for lifting material from floor i to floor j

4.3. Decision Weight (W_n) of Material

Although each major activity has unique materials, the materials can be resolved into an evaluative weighting process. To decide the weight of a material, a layout planner is required to determine the classes of action with respect to each criterion by prioritizing. Decision weights are classified into five important categories, which are described in Table 1. To methodically evaluate these decision weights, a very experienced layout planner is required to score the factors have a significant impact on the decision weight. To accomplish this, the concept of the pairwise comparisons method is adopted.

4.4. Pairwise Comparisons

For the method of Pairwise Comparisons, each material (or alternative) is matched one-on-one with each of the other materials. In this pairwise comparison, there are three ratings-better, the same, and worse important. Each alternative gets 1 point for a one-on-one more important, a half a point for the same, and zero points for less important. For example, a material X which has 1 is more important than a material Y which has 0.5 in terms of a certain activity (Table 2).

After building table, all the values of the indicators in each row of the pairwise comparison are summed. The results are multiplied with weights of all decision then rearranged in descending order with respect to material F_n.

Table 1. Checklist of Decision Weight of Material

Decision Weight (W_n)	Description
W_1	Major activity requirement
W_2	Building space
W_3	Floor access limitation
W_4	Material shape, size, and height
W_5	Safety

Table 2. Pairwise Comparisons Output Matrix

For W_1	Values				
Materials	1	2	3	4	5
1	$ab_{11}=0.5$	$ab_{12}=0$	$ab_{13}=0.5$	$ab_{14}=1$	$ab_{15}=0.5$
2	$ab_{21}=1$	$ab_{22}=0.5$	$ab_{23}=0.5$	$ab_{24}=0.5$	$ab_{15}=0$
3	$ab_{31}=0.5$	$ab_{32}=0.5$	$ab_{33}=0.5$	$ab_{34}=0.5$	$ab_{35}=0$
4	$ab_{41}=0$	$ab_{42}=0.5$	$ab_{43}=0.5$	$ab_{44}=0.5$	$ab_{45}=0$
5	$ab_{51}=0.5$	$ab_{52}=0$	$ab_{53}=1$	$ab_{54}=1$	$ab_{55}=0.5$

5. Representation of a Material Layout

A multiple-floor material layout program assumes that each floor of a building for materials is divided into a grid of unit squares. The size of each grid is variable depending on the building size, but it should be larger than the size of the smallest material. The locations of the vertical handling lifts are pre-specified by the experienced layout planner. Generally, lifts do not occupy any significant space. If a lift occupies significant space in the building, a fixed dummy is assigned at the lift location to prevent the assignment of any other material at that grid. Also, the number of dummies would be assigned to unavailable space.

5.1. Generation of the Initial Population

To generate the ordered set of materials, it is necessary that the area feasibility constraint not be violated, i.e. the sum of the area of all materials on a floor must not exceed the total area available on that floor, including any dummies. For example, there is a building which has two floors, with 5’×6’ grids for each floor (Figs. 3 and 4). For this problem it is shown that activity a_1, a_2, and a_3 (e.g. a_1 may classify the installation of drywall) represented by dark squares are executed on floor 1 and floor 2. There are two square dummies on floor 2 and one stairway at the bottom right corner on each floor also that is also a dummy. A lift, the solid dark box, is located on the bottom side of each floor. It indicates the vertical material transfer method.

The following algorithm is applied to generate the initial population.

Step 0. *Initialize*: Duplicate the blank population that includes dummies and permanent materials as its number of activities. Set the list of activities to be assigned $A = \{a_1, a_2, a_3, ..., a_n\}$.

Step 1. *Choose an activity a_i randomly from A*: Randomly choose and remove a_i from A.

Step 2. *Equal-Distance Sets*: Generate equal-distance sets that are the sets of grids that have the same average distance from a_i.

Step 3. *Assignment of Materials*: Fill the equal-distance sets with materials from the shortest distance to the longest distance according to the weight between a material and a_i.

Step 4. If A is empty, go to step 1. Otherwise, finish the generation of the initial population.

5.2. Equal-Distance Set

An equal-distance set (S) consists of grids that are at the same distance from an activity. If there is more than one place where the same activity occurs, average distance to those places is employed to determine the distance from that activity. For example, if there is an activity at two different grids such as (0, 0, 0) and (4, 4, 0), seven equal-distance sets are generated such at (1, 1, 0), (2, 2, 0), and (3, 3, 0) as an equal-distance set, (0, 1, 0), (1, 0, 0), (0, 3, 0), and (3, 0, 0) as another equal-distance set and so on (Fig. 2).

The vertical process is transferred into the horizontal problem during calculation of the distance since the cost factor is multiplied by horizontal distance. The Fig. 3 and 4 are an example for the above steps. Each activity has seven materials and the weighting of materials for each activity is shown in Table 3.

5.3. Crossover and Mutation

After a population is initialized, crossover and mutation generate a new population for the next generation. The following algorithm is applied to crossover and mutation.

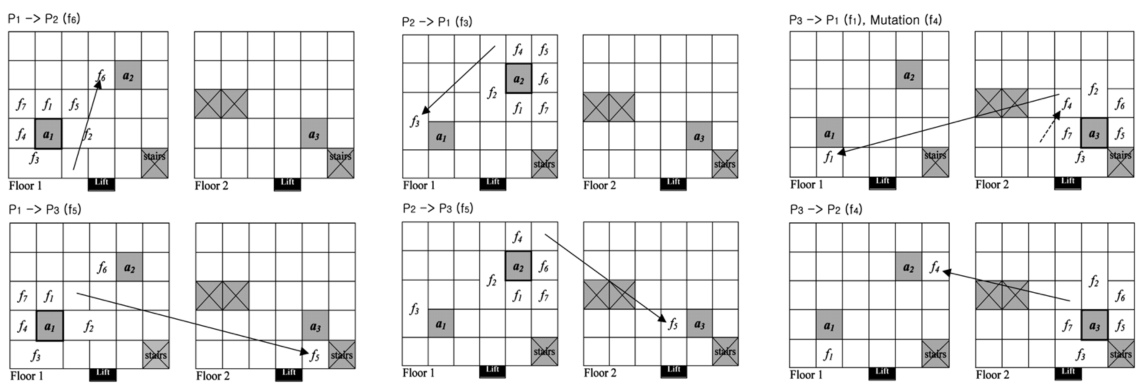
Step 0. *Initialize*: Set a list of populations $P = \{p_1, p_2, p_3, ..., p_n\}$. Set two lists of populations E and Q which are initially empty.

4	4	3.56	3.24	3	
3	3.56	3.16	2.92	2.82	3
2	3.24	2.92	2.82	2.92	3.24
1	3	2.82	2.92	3.16	3.56
0		3	3.24	3.56	4
	0	**1**	**2**	**3**	**4**

■ an activity on floor 0

Numbers are the average distance from that grid to grids where activity

Fig. 2. Equal-distance Sets



(a) (b) (c)

Fig. 3. The First Crossover of P_1, P_2, and P_3

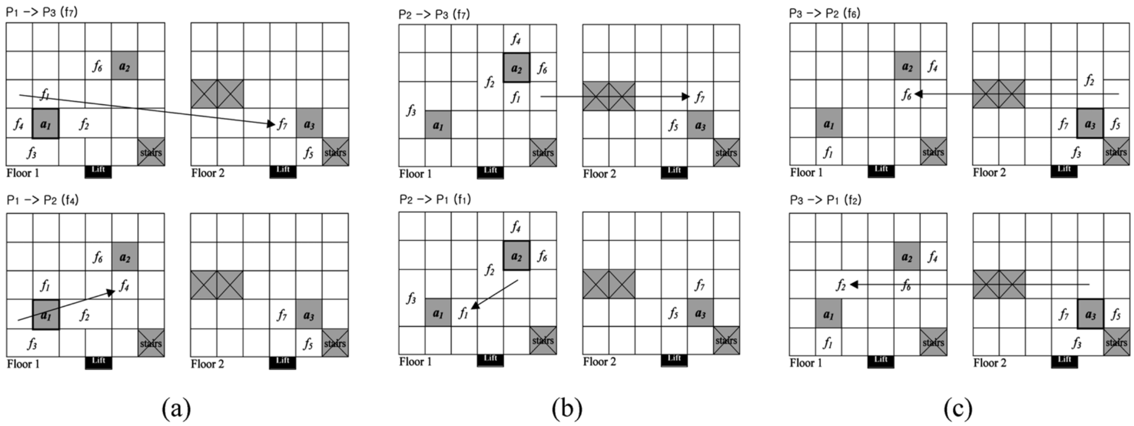


Fig. 4. The Second Crossover P_1, P_2, and P_3

Table 3. Material Priority Ordering

Activity	Ordered Priority Order						
	← High weight				Low weight →		
a_1	f_1	f_2	f_3	f_4	f_5	f_6	f_7
a_2	f_2	f_6	f_4	f_1	f_5	f_7	f_3
a_3	f_3	f_5	f_2	f_7	f_6	f_1	f_4

Step 1. *Randomly select a population p_i from P*: Remove p_i from P and assign p_i to Q in the same location that they appear in p_i.
Step 2. *Randomly select another population p_j from P*: Assign p_j randomly selected from P to Q in the same location that they appear in p_j and remove them from P.
Step 3. *Choose a material*: Select a material which has biggest difference in weight to P and Q
Step 4. *Get the equal-distance set S from p_j*: Remove the material f from p_i and rearrange the materials at removed f in p_i. Remaining materials are relocated from shortest distance to longest distance according the decision weight between materials and the activity in p_i. After rearrange the materials, insert f to p_i following the order of the equal-distance set S.
Step 5. If P is empty, insert p_i to E and go to step 6, otherwise go to step 2.
Step 6. If $E=Q$ go to End, otherwise go to step 7.
Step 7. Remove all populations in Q and add them to P and go to step 1.

5.4. Example of Crossover and Mutation

To illustrate the above algorithm, this paper represents figures by steps. Fig. 3 shows the first crossover and mutation of each population respectively the step 1, 2, and 3.

After the first crossover check step 5, to determine if P is empty, insert p_i to E and go to step 6, otherwise go to step 2. For this example, P is not empty. Thus, Fig. 4 shows the run of the second crossover and mutation of each population respectively, for steps 2, 3, and 4.

After two iterations of crossover and mutation, all populations are not improved by another crossover and mutation. Thus, the result is selected among the populations, which is shown above Fig. 4(a). Also Table 4 shows the selected materials, indicated by the dark sketch.

Table 4. The Result

Activity	Ordered Priority Order						
	← High weight				Low weight →		
a_1	f_1	f_2	f_3	f_4	f_5	f_6	f_7
a_2	f_2	f_6	f_4	f_1	f_5	f_7	f_3
a_3	f_3	f_5	f_2	f_7	f_6	f_1	f_4

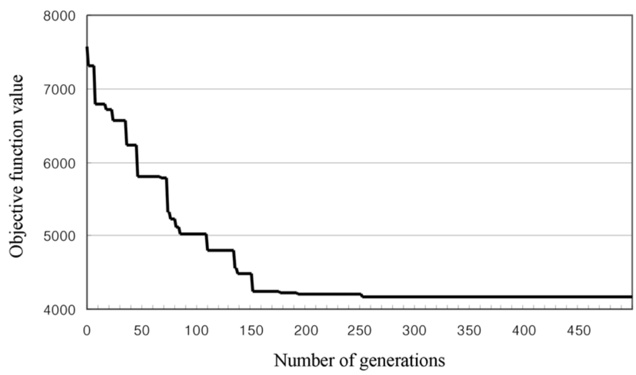


Fig. 5. Plot of Objective Function vs. Generations

6. Experiment of the Number of Generations

A Java program was developed to experiment and implement the Genetic Algorithm (GA) system. Based upon the literature review, it is well known that the number of generation of GA is a very critical parameter for both the efficiency of the search and the performance of the GA. For the experiment of the number of generations, for both the efficiency of the search and the performance of the GA, a range zero to five hundred generations was chosen. An eight-story building with seven activities, twenty-two temporary materials, and two periodic materials was created for the experimentation of the initialization for the program. Also, a dummy and lift were set up for experimentation. Fig. 5 shows a plot of minimum values of the cost function against the number of generations. It can be seen that there is no significance in either values after the 250th generation.

7. Summary

Genetic Algorithm (GA) modeling assumptions are made in order to properly allocate space for materials that will result in lower computational costs and increases in productivity. The following conclusions were made based on the work; (1) the construction material layout problem on high-rise buildings can be represented successfully using the GA modeling described in this paper, (2) GA modeling can be applied to generate workable layouts if an objective function is based on quantitative parameters. Although the GA model has been successfully running, it has to be considered on activities and time relations. Because the model shows a snap shot of floor to floor space plan once the program running. To improve the GA model it is recommended for further development of the model can be extended to evaluate the effectiveness of the overall construction materials layout planning and to manage activities and time relations.

References

Ahuja, L.N. and Nandakumar, V. (1984). "Enhancing Reliability of Project Duration Forecasts." *American Association of Cost Engineers Transactions*, E.6.1-E.6.12.

Emad, E., Tarek, H., Abdel, H.H., and Adel, E. (2001). "Schedule-dependent evolution of site layout planning." *Construction Management and Economics*, Vol. 19, pp. 689-697.

Gero, J.S. and Kazakov, V.A. (1997). "Learning and re-using information in space layout planning problems using genetic engineering." *Artificial Intelligence in Engineering*, Vol. 11, No. 3, pp. 329-334.

Kuntz, K. (1994). "A Construction Crew Evaluation Model." Computer Integrated Construction Research Program, Department of Architectural Engineering, Pennsylvania State University, University Park, PA.

Li, H. and Love, P.E.D. (2000). "Genetic search for solving construction site-level unequal-area facility layout problems." *Automation in Construction*, Vol. 9, No. 2, pp. 217-226.

Lin, K.L. and Haas, C.T. (1996). "An interactive planning environment for critical operations." *Journal of Construction Engineering and Management-Asce*, Vol. 122, No. 3, pp. 212-222.

Oglesby, C.H., Parker, H.W., and Howell, G.A. (1989). *Productivity Improvement in Construction*, Mc Graw-Hill Inc., New York, N.Y.

Philip, M.l., Mahadevan, N., and Varghese, K. (1997). "Optimization of construction site layout - A genetic algorithm approach." *Proceedings of the Fourth Congress held in conjunction with A/E/E SYSTEMS '97*, New York, pp. 710-717.

Rad, P. (1980). "Analysis of working space congestion from scheduling data." *American Association of Cost Engineers Transactions*, F.4.1-F.4.5.

Riley, D.R. (1994). *Modeling the space behavior of construction activities*, Pennsylvania State University, University Park, PA.

Sanders, S.R., Thomas, H.R., and Smith, G.R. (1989). "An Analysis of factors affecting labor productivity in masonry construction." PTI #9003, Pennsylvania State University, University Park, PA.

Tam, C.M., Tong, T.K.L., Leung, A.W.T., and Chiu, G.W.C. (2002). "Site layout planning using nonstructural fuzzy decision support system." *Journal of Construction Engineering and Management-Asce*, Vol. 128, No. 3, pp. 220-231.

Thabet, W.Y. and Beliveau, Y.J. (1997). "SCaRC: Space-constrained resource-constrained scheduling system." *Journal of Computing in Civil Engineering*, Vol. 11, No. 1, pp. 48-59.

Tommelein, I.D., Levitt, R.E., Hayes-Roth, B., and Confrey, T. (1991). "SightPlan experiments: Alternate strategies for site layout design." *Journal of Computing in Civil Engineering*, Vol. 5, No. 1, pp. 41-63.

Tommelein, I.D. and Zouein, P.P. (1993). "Interactive Dynamic Layout Planning." *Journal of Construction Engineering and Management-Asce*, Vol. 119, No. 2, pp. 266-287.

Yeh, I.C. (1995). "Construction-Site Layout Using Annealed Neural-Network." *Journal of Computing in Civil Engineering*, Vol. 9, No. 3, pp. 201-208.

Zouein, P.P., Harmanani, H., and Hajar, A. (2002). "Genetic algorithm for solving site layout problem with unequal-size and constrained facilities." *Journal of Computing in Civil Engineering*, Vol. 16, No. 2, pp. 143-151.

Zouein, P.P. and Tommelein, I.D. (2001). "Improvement algorithm for limited space scheduling." *Journal of Construction Engineering and Management-Asce*, Vol. 127, No. 2, pp. 116-124.

(Received on October 27, 2003 / Accepted on May 12, 2004)