Computer-Aided Site Layout Planning

Farnaz Sadeghpour[1]; Osama Moselhi, F.ASCE[2]; and Sabah T. Alkass, F.ASCE[3]

Abstract: This paper presents an interactive computer-aided site layout model to support site planning in a computer-aided design (CAD) environment and expands upon a model presented earlier by the writers. The developed model performs its task at two levels: Site representation, and site space analysis and allocation. The site representation is carried out using an open architecture supported by object-based concepts. The model offers three tiers of objects: (1) site objects, (2) construction objects, and (3) constraint objects. This structure facilitates the creation of new objects and reuse of domain knowledge, which allows for the gradual expansion and enrichment of the model's knowledge base. At the space analysis and allocation level, the model introduces a geometric reasoning approach to analyze site space for finding an optimum or near-optimum location for facilities. This feature facilitates easy visualization of the site planning process and encourages user participation. The model is structured in three main modules: Database, Project Module, and Layout Control Module. The functionality of each module, along with their interconnectivity is described. The model is implemented using Visual Basic for Applications in AutoCAD environment and Microsoft Access. A numerical example of an actual site layout is presented to illustrate the functionality of the developed model.

DOI: 10.1061/(ASCE)0733-9364(2006)132:2(143)

CE Database subject headings: Construction industry; Computer aided design; Decision support systems; Spatial analysis; Site preparation, construction; Site evaluation.

Introduction

Space on construction sites is recognized as a resource that is as important as other resources of money, time, material, labor, and equipment (Tommelein et al. 1992b; Hegazy and Elbetagi 1999). A well-planned site can contribute to decreasing construction cost and time by minimizing travel time, decreasing time and effort spent on material handling, increasing productivity, and improving safety. However, site visits and interviews with superintendents and site managers reveal that site layout planning is often ignored in the planning phase of construction projects. In practice, space allocation on construction sites is typically carried out on a first-come-first-served basis and mainly through human judgment (Tam and Tong 2003), which could result in chaotic sites and may give rise to productivity losses and safety-related incidents.

Due to the complexity and the number of factors involved, computers were identified as an efficient tool to help site planners in their task. Over the last few decades several computer-based site layout systems have been developed. *CORELAP* (Moore 1971) was one of the first computer models developed for plant layout using operations research techniques. Later, artificial intelligence (AI) emerged as a viable tool to analyze site layouts and provide near-optimum solutions. Different AI techniques—such as knowledge-based systems (Hamiani 1987; Tommelein et al. 1991, 1992b; Flemming and Chien 1995; Choi and Flemming 1996), neural networks (Yeh 1995), and genetic algorithms (Philip et al. 1997; Li and Love 1998; Hegazy and Elbeltagi 1999; Harmanani et al. 2000; Mawdesley 2002; Zouein et al. 2002; Tam and Tong 2003)—were explored in the development of site layout systems. The main advantage of AI methods lies in their ability to deal with inexact, incomplete, and/or ill-structured problems (Luger and Stubblefield 1998). Other models have been developed using a hybrid of techniques. MoveSchedule (Tommelein and Zouein 1993a,b; Zouein 1996) is a dynamic space scheduling system that incorporates expert's knowledge in an optimization process. Cheng (1992) integrates a knowledge-based approach in a model using a geographic information system. However, to date no standard tool has gained wide acceptance by industry. This has been attributed to the difference between how domain experts address the problem and the way computer-based systems represent it (Tommelein et al. 1992a). In an effort to bridge this gap, researchers have suggested the use of knowledge-level description that is more abstract and code-independent (Balkany et al. 1991). This description includes the structure, functions, and properties of the components of the model. Such description is particularly important for site layout product models since there are few of such models that contain the domain knowledge. Further, the existing product models do not support the reuse of the knowledge and experience utilized in previous projects (Tommelein et al. 1992a). In general, previously developed site layout models exhibit one or more shortcomings of limited optimization objectives, rigid problem setup factors (e.g., facilities, site bound-

[1]Faculty, Dept. of Building, Civil, and Environmental Engineering, Concordia Univ., 1819 de Maisonneuve West, Montreal QC, Canada H3G 1M8.

[2]Professor, Dept. of Building, Civil, and Environmental Engineering, Concordia Univ., 1819 de Maisonneuve West, Montreal QC, Canada H3G 1M8 (corresponding author). E-mail: moselhi@cbs-engr.concordia.ca

[3]Professor, Dept. of Building, Civil, and Environmental Engineering, Concordia Univ., 1819 de Maisonneuve West, Montreal QC, Canada H3G 1M8.

Note. Discussion open until July 1, 2006. Separate discussions must be submitted for individual papers. To extend the closing date by one month, a written request must be filed with the ASCE Managing Editor. The manuscript for this paper was submitted for review and possible publication on May 21, 2004; approved on June 17, 2005. This paper is part of the ***Journal of Construction Engineering and Management***, Vol. 132, No. 2, February 1, 2006. ©ASCE, ISSN 0733-9364/2006/2-143–151/$25.00.

J. Constr. Eng. Manage., 2006, 132(2): 143-151

ary, site conditions), mismatch between user's and systems' approach, lack of user-system interaction, and lack of knowledge recycling tools (Sadeghpour et al. 2004b).

This paper presents a site layout model designed to overcome the limitations cited above. The model addresses these limitations at two levels. At the first level, it tackles data structure and knowledge representation. Site layout-related elements are systematically identified and organized into object libraries. This classification assists in formalizing the representation of site layout problems in a simple format that is comprehensible by practitioners. An open architecture approach is proposed to support the generation of new objects in accordance with the diverse nature of construction sites. Such a characteristic renders the model flexible at the problem setup phase, not only in formation of objects, but also in defining new constraints to satisfy different objectives. Thus, it facilitates user contribution to the expansion and refinement of the knowledge base of the model, which enriches the model's capabilities. As well, it assists in recycling the domain knowledge utilized in current projects for later use on future projects. At the second level, the model performs space analysis and allocation of objects using a geometric reasoning approach for analyzing the site space. This approach resembles the human reasoning process and addresses the gap reported above, between the way domain experts develop site layouts and the way computer-based models approach it. In addition, the proposed model, through its open architecture, engages users in decision making at different stages. The developed model is implemented in computer-aided design (CAD) environment to facilitate users' visualization, comprehension, and interaction.

Proposed Model

Object-based concepts are utilized to formalize the data structure for the objects that represent the site conditions and the temporary facilities to be placed on the site. Based on this formalization, an open-architecture project setup module is developed. This architecture allows for creation and addition of new objects in each category at the user level. The newly created objects are captured by the model and remain in its database for future use. This allows users to populate the model database with new objects and tailor it to their respective work environment. Further, a geometric space analysis methodology is developed to find the optimum or near-optimum location for each object on site. The proposed methodology is applied to place a set of *construction objects* on site, while respecting the presence of a set of *existing objects on site* and satisfying a set of locating *constraints* defined by the user. As such, the site layout elements are clustered into three groups of objects: (1) *site objects*; (2) *construction objects*; and (3) *constraint objects* (Sadeghpour et al. 2002, 2003). Fig. 1 illustrates the general structure of the developed model and the relation between its three basic components: Database, Project Module, and Layout Control Module. Each of the three groups of objects is defined in respective submodules within the Project Module (i.e., Site Module, Construction Module, or Constraint Module), and stored in its respective library in the model's database (i.e., site library, construction library, or constraint library). All objects defined for a given project are then passed to the Layout Control Module, which conducts the search to find the optimum location for each construction object on site. The modules are designed to perform and interact with the user in CAD environment. The project setup phase of the proposed model has been previously introduced (Sadeghpour et al. 2004b). The following two sections, however, provide additional detailed description of its structure and functionality. Further, in the Layout Control Module section, a new methodology for site space analysis is introduced.

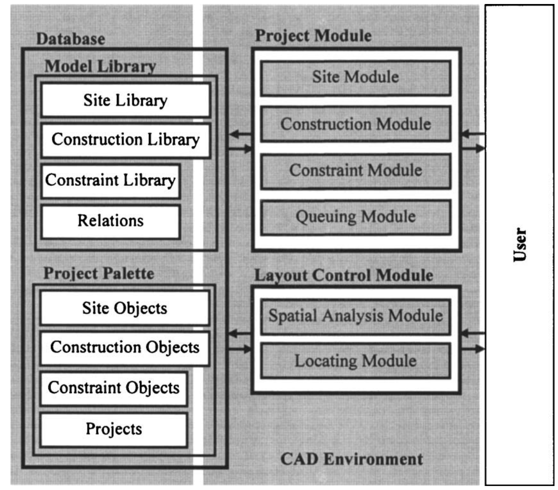

Fig. 1. Model structure

Database

Object-based concepts were utilized to represent the three aforementioned tiers of objects. Object-based approach strongly promotes the formalism of data typing and encapsulation of information, and hence inherently facilitates regeneration of the objects used in the model. Contrary to the object-oriented approach, no explicit inheritance scheme is imposed in the object-based approach (Zamanian 1992). The proposed tiers are designed as object classes, encapsulating their relevant properties. These properties include geometric and non-geometrical data and knowledge. The geometric information of an object can be accessed from a drawing file that contains a graphical instance of that object. The built-in database of the CAD system stores the geometrical and graphical properties of the object, such as area, perimeter, color, and line type. The nongeometrical properties are stored in a relational database, external to the CAD system. Fig. 2 depicts the relationships among the entities of the database. The database is divided into two functional sections: (1) the Model Library (ML) that acts as an object gallery from which the objects can be viewed and selected, and (2) the Project Palette (PP) that keeps record of the objects defined to represent the project at hand. The database consists of eight entities: "Site Library," "Construction Library," "Constraint Library," and "Relations" form the Model Library; where "Site Palette," "Construction Palette," "Constraint Palette," and "Projects" compose the PP.

When a construction or site object is selected from the library, an instance of its geometry is generated on the screen. The "file path" property provides the location of the drawing file, which can be inserted into the screen as a block. Graphical objects have a unique identification property, called *handle*, that is consistent through the life of that object and hence is the best reference to address an object. This property is assigned to the graphical objects upon their creation/insertion on the screen and can be accessed from the built-in database of the CAD system. When a site or construction object is selected from the Model Library to be added to the project, its handle is recorded in the respective table of the PP (i.e., site palette or construction palette). In other words,

J. Constr. Eng. Manage., 2006, 132(2): 143-151

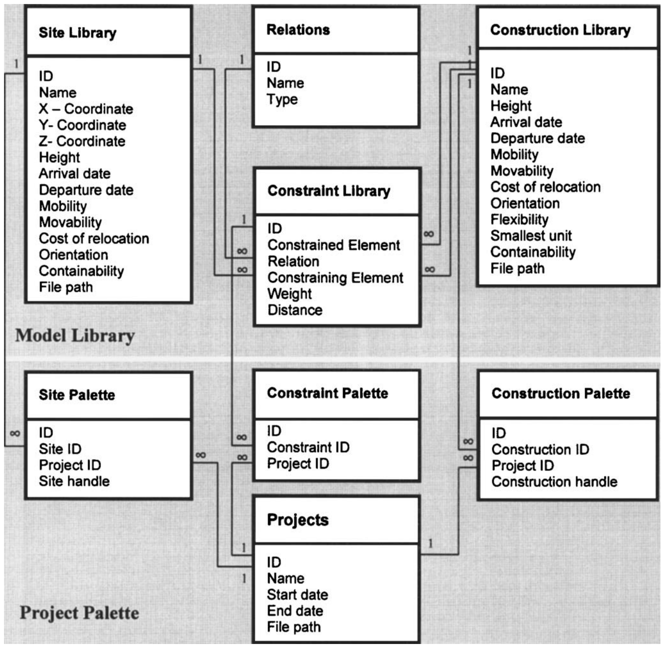

Fig. 2. Relationship diagram of the database

the handle property is the overlap between the built-in database of the CAD system and model's database to ensure connectivity between the two. As such, for each object selected for the project at hand, its nongeometric properties are coupled with the geometric ones read from the built-in database of the CAD system. By knowing the handle of an object, its geometrical information, such as area, perimeter, and/or centroid, can be obtained from the built-in database of the CAD system. Conversely, through the handle property, the nongeometrical properties of an object can be retrieved, modified, and saved from within the CAD system.

Project Module

The design of a site layout project commences at the Project Module, where the constructing elements of the project are defined in each of the respective submodules: Site Module, Construction Module, and Constraints Module, and are lined up in the Queuing Module in the order they are to be placed on site [Fig. 3(a)].

The site module starts by defining the boundaries of the project site. Once the site boundary is set, the site and construction modules prompt the user to *define* site objects and construction objects, respectively. The term *define* denotes two functions in the proposed model: *selecting* objects from the respective library, and *creating* new objects. Each of the site library, construction library, and constraint library, offers a set of related objects from which the user can *select* objects most suited to represent the project at hand. The selected objects will be added to the PP, which stores project-related data. If a desired object is not available from the libraries, the user is provided with tools to *create* it. As such, the model provides an open architecture for setting up a project. Newly created objects are added to the PP and the Model Library for future use. This allows planners to apply their individual problem solving strategies, and thus, directly contribute to the enrichment of the database of the model (Sadeghpour et al. 2003).

When *creating* an object, the model first prompts the user to define the geometry of that object. Depending on the case, this can be carried out using: (1) the readily available geometry of that object, which is directly extracted from a drawing file; (2) model-assisted drawing environment, where coordinates of the corners of the object's geometry are given as text input and the model generates the geometry of the object; or (3) the CAD system, which allows for drawing directly on the work area [Fig. 3(b)]. Once the geometry of an object is generated, the model then prompts the user to input its nongeometric data. The nongeometric data acquisition is conveyed in a text format questionnaire, and stored in the model's database [Fig. 3(c)].

The constraint module defines the interconnectivity among site and construction objects using a set of constraint objects. These objects are rules and preferences defined for locating each construction object. Each constraint object is assigned a proximity weight to indicate its importance with respect to other constraints. Fig. 4 shows the structure of a constraint object consisting of three parts: (1) a construction object for which the constraint object is being defined, referred to as the *constrained element*; (2) a topographic *relationship*, such as "as close as possible to," "west of," "visible from," and "within x meters of"; and (3) a *constraining element*, which can be a construction object or a site object. The developed structure facilitates the creation of new constraint objects in accordance with the open architecture design of the proposed model. A set of constraint objects is assigned to each construction object in the project palette. The construction object, for which the constraint is being defined, forms the *constrained element*. The *topographical relationship* is selected from a list provided by the Model Library. The *constraining element* can be selected from a list of construction and site objects in the PP. More details on the structure of objects and project setup can be found in Sadeghpour et al. (2003, 2004b).

Graph theoretic techniques have been applied earlier by others to decide on adjacency between pairs of objects (Foulds 1983, Foulds et al. 1985). Two other important placement methods are improvement and construction methods (Moore 1980). Improvement methods start from an initial layout and generate alternative layouts by changing the position of objects. Each layout is then evaluated against a predefined objective function and the layout that best satisfies the objective function is identified. Construction methods, used in the proposed model, add one object at a time to the layout, considering the updated status of the site before adding each object. In this case, the order in which construction objects are located on site affects the final layout. Tam et al. (2002) have applied fuzzy logic to evaluate the order of objects based on pairwise comparisons. Zouein (1996) suggested random selection of objects or the use of one of the three heuristics: "area requirement," "duration on site," or "cost of relocation," without articulating a formal structure for the collective application of these heuristics. However, it is important to ensure that the planner is able to queue objects based on experience, intuition, and due considerations of applicable site constraints. In an effort to achieve such an open, dynamic, and user-interactive process in the proposed model, assignment of priorities is established based on a multiattribute queuing score (Q). This score accounts for a combination of weighted heuristics, such as those referred to earlier (i.e., area requirement, duration on site, cost of relocation), as well as other factors such as "Number of constraints," "Total weight of constraints," "Time of arrival," and "Number of *constrained elements.*" The latter refers to the frequency of the object being selected as a *constraining element* in a constraint object. The planner selects applicable attributes to be used in the queuing process of each project and assigns a relative weight to each to indicate its importance compared to others (see Fig. 5). The mul-

J. Constr. Eng. Manage., 2006, 132(2): 143-151

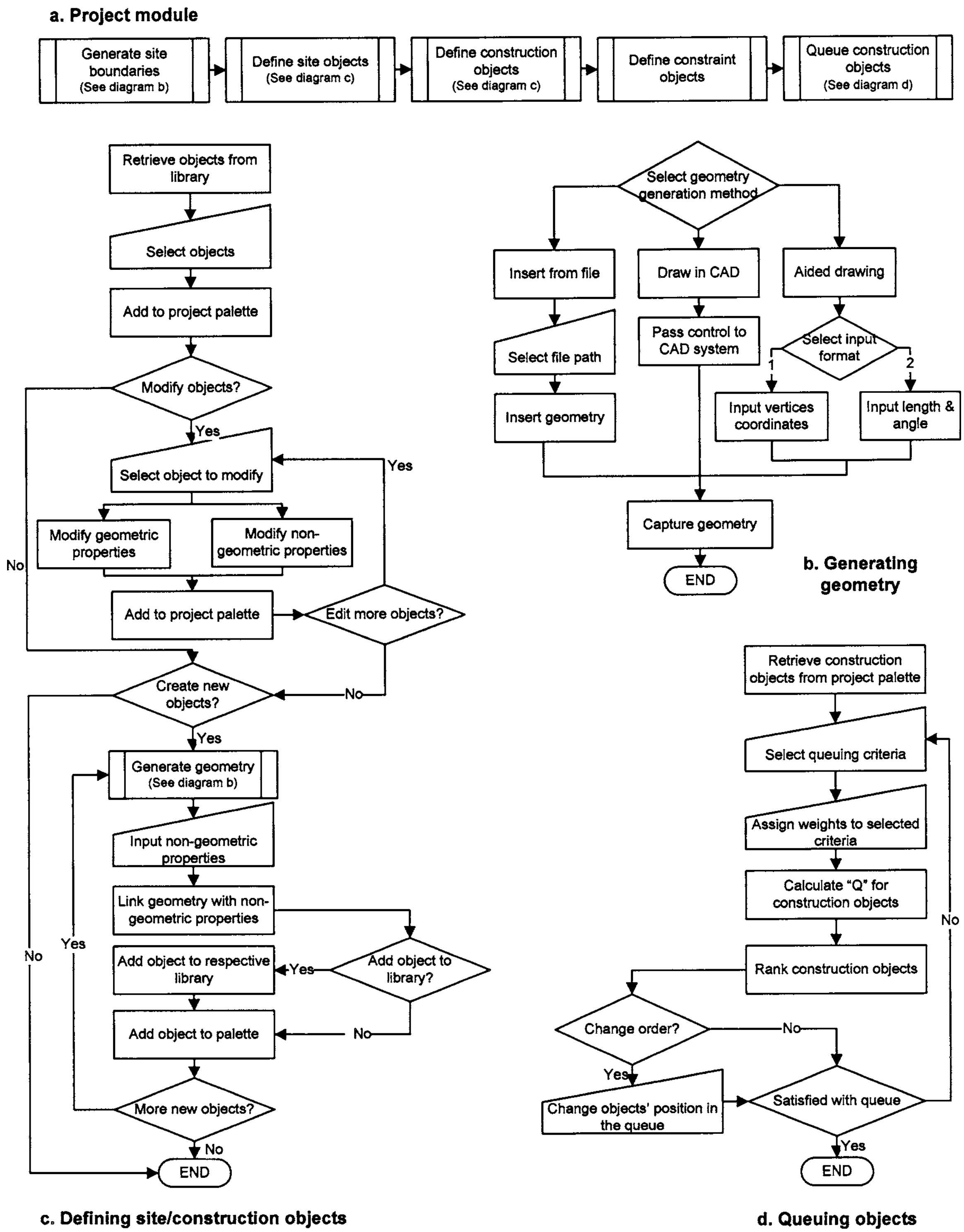

Fig. 3. Dataflow in the project module

tiattributed queuing score (Q) for each construction object is calculated as

$$Q = \sum_{1}^{n} \frac{C_i}{C_{\text{Max}}} \text{[00e2][0088][0099]} Wc_i \qquad (1)$$

where C_i=value associated with the ith attribute; C_{Max}=maximum value of that attribute for all objects; and Wc_i=weight assigned to this attribute. Accordingly, objects are queued in descending order of their queuing score. The planner

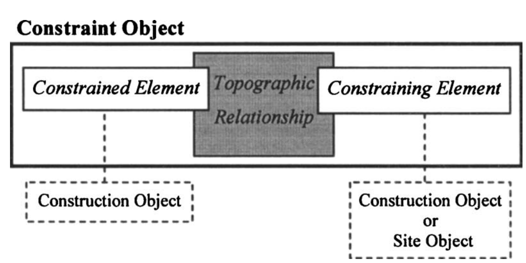

Fig. 4. Structure of a constraint object

J. Constr. Eng. Manage., 2006, 132(2): 143-151

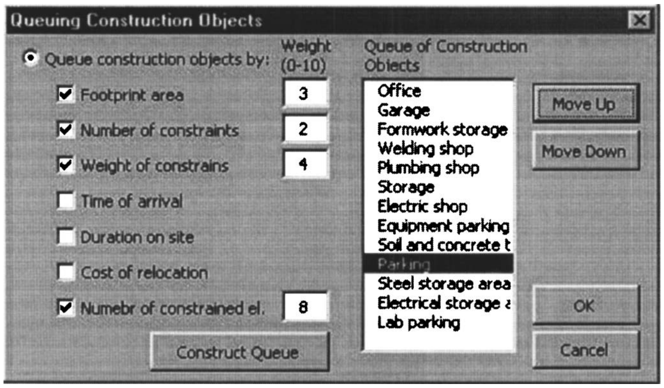

Fig. 5. Queuing of construction objects

can choose to proceed with that queue or modify the order of its objects [Fig. 3(d)]. As well, different combinations of the attributes leading to different orders of objects can be invoked to study the impact on the final layout.

The procedure described above is applied for queuing 13 construction objects presented in Table 1. Fig. 5 depicts a sample screen for the queuing process in the developed model. The four attributes selected in Fig. 5 are used to form the queuing criteria. Each is assigned a weight to signify its importance to the planner in assigning the space on site. The data associated with the attributes selected for this case are summarized in Table 1 (Columns 3 to 6). It should be noted that these data were retrieved directly from the model's database, upon forming the queuing criteria. Table 1 also contains the calculated queuing score (Q) and the rank of each object in the queue.

Layout Control Module

The queue of construction objects is passed to the layout control module to place the objects on site. This module performs two major tasks: Analyzing site space to find the optimum or near-optimum location for each construction object in the queue, and

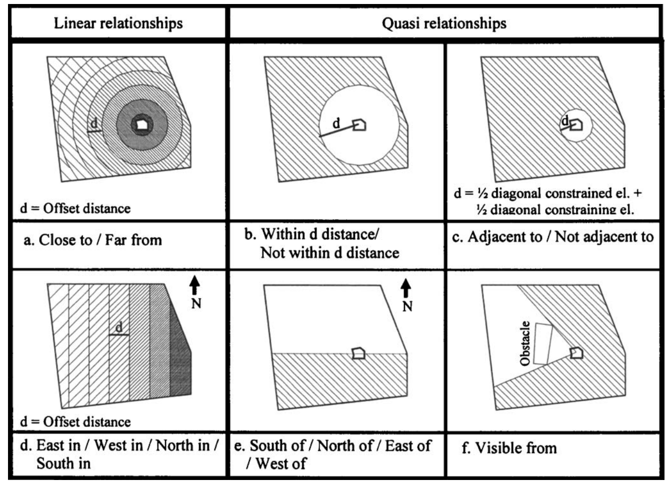

Fig. 6. Geometric representation for six types of relationships

placing the construction objects on site. Each of these tasks is performed by "spatial analysis" and "locating" submodules, respectively.

The spatial analysis submodule scans the available site area to find the best location based on the knowledge and information provided by the PP. The process of finding the location is done through a geometric reasoning search (Sadeghpour et al. 2004a). When locating a construction object, the model retrieves the constraint objects assigned to it. For each constraint defined for the construction object at hand, the site is divided into segments. Each site segment represents a different degree of satisfaction for that constraint. The discretization method differs according to the type of topographical relationship used to define the constraint. Fig. 6 shows the discretization method for six types of relationships. These relationships are obviously not exhaustive, but are sufficient for defining most of the cases. These relationships are divided into two groups: (1) linear relationships; and (2) quasi-relationships. Linear relationships represent a gradual transformation in the satisfaction score of each site segment, whereas

Table 1. Data Pertaining to Queuing of Objects

		Queuing attributes					
Number	Construction object	Area (m^2)	Number of constraints	Weight of constraints	Number of constrained elements	Q	Rank
1	Office	500	7	560	8	14.23810	1
2	Parking	750	2	160	0	2.071429	10
3	Garage	750	2	130	5	6.857143	2
4	Storage	320	5	340	0	4.009524	6
5	Welding shop	150	3	200	3	5.357143	4
6	Plumbing shop	600	2	130	3	4.785714	5
7	Electric shop	150	2	150	2	3.714286	7
8	Formwork storage area	6,300	3	190	1	6.214286	3
9	Steel storage area	100	2	150	0	1.690476	11
10	Electrical storage area	200	1	100	0	1.095238	12
11	Equipment parking	1,200	4	260	0	3.571429	8
12	Soil and concrete test labs	144	3	190	1	3.282857	9
13	Lab parking	90	1	100	0	1.042857	13
Max value (C_{Max})		6,300	7	560	8		
Weight (Wc)		3	2	4	8		

J. Constr. Eng. Manage., 2006, 132(2): 143-151

quasi-relationships divide the site in two segments and demonstrate a strict preference of one segment over the other (Brans et al. 1986). For example, "close to" or "far from" relationships are represented by means of circles circumscribing the *constraining element*, dividing the site into ring-shaped regions. Assuming the area inside a ring has the same distance from the *constraining element*, each ring represents a different degree of satisfaction for the closeness or farness relation (see Fig. 6).

As geometric representation for all the *constraint objects* associated with the construction object in hand are generated, the site is divided into smaller and smaller segments resulting from intersections of various geometrical representations as will be shown later in the case example. Considering the fact that the intersection of two areas contains the attributes of both, the satisfaction score for the area of intersection is calculated as the sum of the scores associated with the two intersecting areas. As such, the utility score (U) for each site segment is expressed as

$$U = \sum_{1}^{n} S_i \,\text{[00e2][0088][0099]}\, W_i \qquad (2)$$

where S_i=satisfaction score for the ith constraint; and W_i=weight assigned to the ith constraint. Based on Eq. (2), the utility score for each site segment is calculated and they are sorted accordingly. The geometric representation of constraints divides the site into a set of irregular segments that are neither of equal size nor of the same shape. The utility function of site segments is calculated at their respective centroids, which form an irregular network of grid points. The width of the rings representing linear relationships, such as farness and closeness, affect the shape and size of this grid.

The size of the solution segment found by the model depends on the offsetting width in linear relationships and indicates the precision of the answer. The smaller the offset width chosen, the smaller the size of the grid, and consequently the more precise the location of the construction object. If higher precision is required, the model allows for refining the analysis within the solution segment. The grid refining process can take place in several iterations, zooming down on the near-optimum location, until the precision of the location, and in other words the size of the solution segment is deemed satisfactory. The planner can accept the location found by the model, or veto the model's solution and select another location based on the analysis and results provided by the model. The grids and geometries resulting from each analysis iteration are stored in a separate drawing layer and can be viewed separately or simultaneously to assist the planner in deciding the final location of the object at hand. If a location other than those ranked by the model is selected, the model can calculate the utility score of that location for comparison purposes.

Once the final location of a construction object is decided, the locating module proceeds with laying the object on the identified location. By default, the model matches the controid of the object in hand with that of the identified segment for the object location. The user then could choose to fine-tune the location of the object as required using corner-to-corner adjacency [Fig. 7(a)], vertical and horizontal alignment [Fig. 7(b)], and/or rotating the object (not shown in Fig. 7). The locating module further verifies whether the object overlaps with an existing object on site to satisfy the nonoverlap condition.

When the object is positioned on site, the layout module deducts its footprint area from the available site area. Once all the construction objects are located, the total utility function (f) of the layout can be calculated as

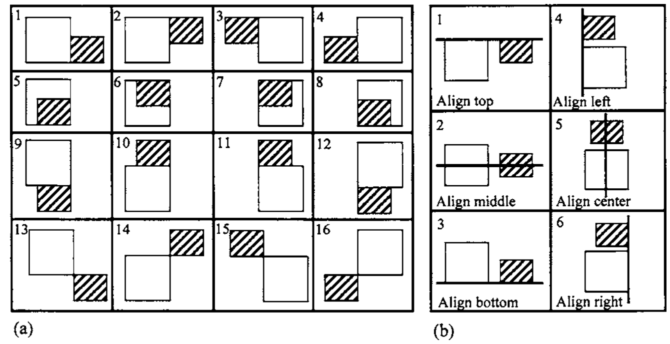

Fig. 7. Alignment of objects: (a) corner-to-corner adjacencies; and (b) vertical and horizontal alignments

$$f = \sum_{1=i}^{n} \sum_{1=j}^{m} S_{ij} \,\text{[00e2][0088][0099]}\, W_{ij} \qquad (3)$$

where S_{ij}=satisfaction rate of ith constraint defined for jth object; and W_{ij}=weight of ith constraint defined for jth object. Eq. (3) is useful in comparing the fitness of different layouts generated by different project setup factors.

Implementation

The model described above, was implemented as proof of concept, using Visual Basic for Applications (VBA) in AutoCAD environment, and Microsoft Access to develop the model's database. VBA is an object-oriented programming environment with development capabilities similar to those of Visual Basic (VB). Unlike VB, VBA runs in the same process environment as AutoCAD, providing seamless integration and fast execution of the program. It also provides integration with other VBA-enabled applications, such as MS Access. This enables AutoCAD to be an automation controller for such applications, using their respective object libraries (Autodesk 2001). As such, VBA provides a seamless link between the main components of the model, supported by a powerful graphical user interface (GUI). The model has been demonstrated to practitioners from the industry at several stages of its design and development and their recommendations were considered at each stage.

Case Example

Case Description

A case study from industry was developed to illustrate the flexibility and capabilities of the developed model. The case involved the design of a layout for the site of a major hydro project, LG-2, constructed in Quebec, Canada. The site has an area of 28'178 m^2 and was required to accommodate the 13 facilities listed in Table 1. Fig. 8 shows the layout of the facilities on site as designed by the project team. The numbers refer to the construction objects listed in Table 1. It should be noted that the site engineer, who was involved in the preparation of the original site layout, was requested to carry out the analysis and generate the layout using the developed model with the assistance of a computer operator. Further, to enable a comparison, the set of constraints used in the original layout were also imposed in the case analysis performed using the developed model.

J. Constr. Eng. Manage., 2006, 132(2): 143-151

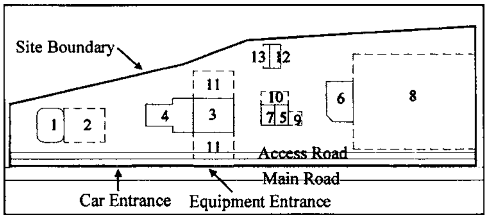

Fig. 8. Actual site layout of LG-2 project

Generation of Layout

To map the presented project in the proposed model, the project module starts with generating the required objects. After generating the boundaries of site, three site objects were identified and their respective representations were generated: Car entrance, equipment entrance, and access road. The geometry of these objects was generated using the features in the model and they were placed on their known location on the site. Thirteen facilities were modeled as the construction objects and added to the PP. For each construction object, a number of constraints were defined. To enable a comparison, the constraints were identical to those considered at the time of actual site design by members of the project team. Table 2 summarizes the constraint objects, their composing elements, and their respective priority weights. Once the PP is set, the Queuing Module ranks the order of construction objects based on four attributes as described earlier in the queuing example (see Fig. 5) and summarized in Table 1.

Upon the queuing of objects, the Spatial Analysis Module is activated to find a location for each object according to the constraints defined for them. At the beginning of site analysis, three regions on the site, highlighted in Fig. 9 as A, B, and C, were recognized as "unavailable." Region A is a 10 m wide space inside the site boundary, provisioned to keep construction objects from being located on the edge of the site. Regions B and C indicate 20 m of clearance around the entrances to the site. As an

Table 2. Constraint Objects Defined for the LG-2 Project

Number	Constrained element	Relation	Constraining element	Weight
1	Office	Far from	Electrical shop	90
		Far from	Welding shop	100
		Far from	Plumbing workshop	90
		Not within *d* of	Garage	50
		Not within *d* of	Access road	50
		Close to	Car entrance	80
		West of	Site	100
2	Parking	Next to	Office	90
		Close to	Car entrance	70
3	Garage	Within *d* of	Equipment entrance	80
		Not within *d* of	Office	50
4	Storage	Close to	Office	50
		Visible from	Office	70
		Next to	Garage	100
		Close to	Car entrance	40
		Close to	Equipment entrance	80
5	Welding shop	Next to	Plumbing workshop	40
		Far from	Office	100
		Not within *d* of	Formwork storage area	60
6	Plumbing workshop	Close to	Welding shop	40
		Far from	Office	90
7	Electrical shop	Close to	Plumbing workshop	60
		Far from	Office	90
8	Formwork storage area	Not within *d* of	Welding shop	60
		East of	Site	80
		Close to	Access road	50
9	Steel storage area	Next to	Welding shop	90
		Far from	Office	60
10	Electrical storage area	Next to	Electrical shop	100
11	Parking lot for equipment	Next to	Garage	100
		Close to	Car entrance	10
		Close to	Equipment entrance	70
		North of	Garage	80
12	Soil and concrete test labs	Close to	Garage	60
		Close to	Electrical shop	80
		North of	Site	50
13	Lab parking	Next to	Soil and concrete test labs	100

Note: d=distance specified by user.

J. Constr. Eng. Manage., 2006, 132(2): 143-151

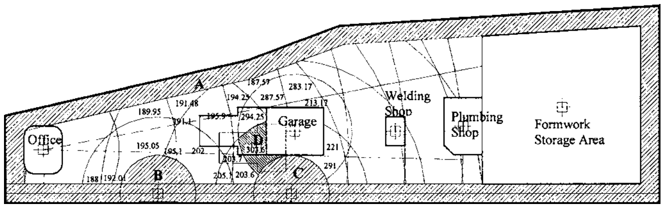

Fig. 9. Site analysis for Storage

example, Fig. 9 shows the site analysis for the "Storage" as the sixth construction object to be laid on the site (see Table 1), considering five constraint objects (see Table 2). These constraints indicate that the storage should be located close to the two entrances to ease deliveries; close to the office since several visits to the storage were predicted, and visible from the office to maintain security. The constraint object specifying the adjacency between the Storage and Garage was specifically created for this project since the site manager intended to use the same structure for the two. When the available site area is analyzed based on the constraints, the segment highlighted as D receives the highest satisfaction score, and accordingly, the object is first placed at that location. It should be noted that the numbers shown in Fig. 9 represent the satisfaction score (U) for a set of site segments closer to the segment D. The locating module further fine-tunes the location of Storage and aligns it with the Garage, invoking the corner-to-corner adjacency feature shown in Fig. 7.

Fig. 10 shows the layout of the site after the analysis is performed for all the construction objects in the queue. The numbers refer to the construction objects listed in Table 1. It is interesting to note the similarities and dissimilarities between the generated layout and the original one. Equipment Parking (11) is moved to one side of the Garage (3) to maintain to clearance area defined for the Equipment Entrance. Also its shape was redefined in accordance with the clearance defined for the site boundary while providing the same area. Similarly, the shape of Formwork Storage Area (8) was slightly resized to take advantage of the unused corner of site and give space on its left, which is in continuity with the open space. Electrical Shop (7) in the proposed model is shifted to the right of the site to better accommodate the two constraints of closeness to the Plumbing Shop (6) and farness from the Office (1). In general, the model was able to generate a layout that is close to the original site arrangement in shorter time. The differences between the generated layout and the original layout can be attributed to the fact that the generated layout better satisfied the defined constraints.

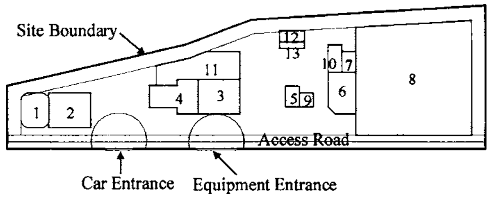

Fig. 10. Site arrangement for LG-2 generated using the proposed model

Discussion

The case example presented here demonstrated the capabilities of the model in overcoming the limitations of other models referred to earlier in the paper. The model was capable of creating project requirements, which consisted of multiple facilities with irregular shapes and a variety of locating constraints. The objects and constraints defined in this project were added to the database and can be recycled for future projects. Due to model's flexibility in defining constraints, the model was able to include optimization objectives other than minimum travel distance, such as security. The geometry-based approach utilized in the presented model, as well as its data flow, facilitate its use by practitioners. Additionally, the open architecture referred to earlier supports the user interaction, and provides users with the opportunity of utilizing their own knowledge and experience.

The model improves the current practice on two points: (1) in the project setup phase, the model provides a central platform that supports users in defining relevant objects and applicable constraints. (2) In the site analysis and allocation phase, the model was able to analyze the constraints defined for each object and find a location in a considerably shorter time compared to the actual time it took site planners. When the analysis was performed on a regular office computer (Pentium IV processor with 2 GHz processor), on average it took the model 30 s to find a near-optimum location for each construction object; whereas the design of the original layout was the product of several meetings of various decision makers.

Summary and Concluding Remarks

This paper presented a CAD-based site layout model designed to aid site planners in developing layouts in an interactive manner. The developed model provides users with a design support tool that helps its users to utilize their knowledge in design of efficient site layouts. The basic components of the model—including its database, Project Module, and Layout Control Module—were described and the functionality of each module was described. The paper discussed a formalized data structure for site layout modeling in detail and introduced a new methodology for site space analysis and allocation. This methodology is geometry-based to support visualization and facilitate user-system interaction. A case example of an actual site layout was analyzed using the developed model. The model was able to generate a layout close to the one created by site planners. The differences between the two layouts are attributed to the degree of satisfaction associated with the constraints considered in the project. The developed model resulted in a layout that better satisfied the specified constraints.

The case example demonstrated that the developed model was capable of overcoming the limitations of the models developed by others. The model has been validated through a number of case studies, including those from literature (Sadeghpour 2004). The model improves current practice by providing site planners and decision makers with a centralized computer environment for defining the required site objects and constraints in a project. As well, it notably reduces the time required for site analysis. To realize a site layout model that is acceptable in the industry, the user interface of the developed model should be further validated through hands-on experiments with participation of industry practitioners.

J. Constr. Eng. Manage., 2006, 132(2): 143-151

Acknowledgment

The writers would like to acknowledge Mr. German Cardinal, practicing consultant, for his interest and for sharing his valuable time and experience in preparing the case study presented in this paper.

References

Autodesk. (2001). *ActiveX and VBA developer's guide*, Autodesk Inc., San Rafael, Calif.

Balkany, C. A., Birmingham, W. P., and Tommelein, I. D. (1991). "A knowledge-level analysis of several design tools." *Proc., Int. Joint Conf. on Artificial Intelligence in Designs, AID91*, Butterworth–Heinemann, Surrey, U.K.

Brans, J. P., Vincke, P., and Mareschal, B. (1986). "How to select and how to rank projects: The PROMETHEE method." *Eur. J. Oper. Res.*, 24(2), 228–238.

Cheng, M. Y. (1992). "Automated site layout of temporary construction facilities using geographic information system (GIS)." PhD dissertation, Univ. of Texas, Austin, Tex.

Choi, B., and Flemming, U. (1996). "Adaptation of a layout design system to a new domain: Construction site layouts." *Proc. of 3rd Congress Computing in Civil Engineering, ASCE*, A/E/C Systems, Anaheim, Calif., 718–725.

Flemming, U., and Chien, S. F. (1995). "Schematic layout design in SEED environment." *J. Archit. Eng.*, 1(4), 162–169.

Foulds, L. R. (1983). "Techniques for facilities layout: Deciding which pairs of activities should be adjacent." *Manage. Sci.*, 29(12), 1414–1426.

Foulds, L. R., Gibbons, G. B., and Giffin, J. W. (1985). "Facilities layout adjacency determination: An experiment comparison of three graph theoretic heuristics." *Oper. Res.*, 33(5), 1091–1106.

Hamiani, A. (1987). "CONSITE: A knowledge-based expert system framework for construction site layout." PhD dissertation, Univ. of Texas, Austin, Tex.

Harmanani, H., Zouein, P., and Hajar, A. (2000). "An evolutionary algorithm for solving the geometrically constrained site layout problem." *Proc., of the 8th Int. Conf. on Computing in Civil Engineering*, ASCE, Reston, Va., 1442–1449.

Hegazy, T. M., and Elbeltagi, E. (1999). "EvoSite: Evolution-based model for site layout planning." *J. Comput. Civ. Eng.*, 13(3), 198–206.

Li, H., and Love, P. E. D. (1998). "Site-level facilities layout using genetic algorithms." *J. Comput. Civ. Eng.*, 12(4), 227–231.

Luger, G. F., and Stubblefield, W. A. (1998). *Artificial intelligence and the design of expert systems*, Benjamin/Cummings, Redwood City, Calif.

Mawdesley, M. J., Al-jibouri, S. H., and Yang, H. (2002). "Genetic algorithm for construction site layout in project planning." *J. Constr. Eng. Manage.*, 128(5), 418–426.

Moore, J. (1971). "Computer program evaluates plant layout alternatives." *Ind. Eng.*, 3(8), 19–25.

Moore, J. (1980). "Computer methods in facility layout." *Ind. Eng.*, 12(9), 82–93.

Philip, M., Mahadevan, N., and Varghese, K. (1997). "Optimization of construction site layout—A genetic algorithm approach." *Proc., 4th Congress on Computing in Civil Engineering*, ASCE, Reston, Va., 710–717.

Sadeghpour, F. (2004). "A CAD-based model for site layout." PhD dissertation, Concordia Univ., Montreal.

Sadeghpour, F., Moselhi, O., and Alkass, S. (2004a). "Geometric reasoning for site space analysis." *Proc., World Building Congress*, CIB, Toronto.

Sadeghpour, F., Moselhi, O., and Alkass, S. (2004b). "A CAD-based model for site planning." *Autom. Constr.*, 13(6), 701–715.

Sadeghpour, F., Moselhi, O., and Alkass, S. (2003). "Open architecture for site layout modeling." *Proc., Int. Symp. of Automation and Robotics in Construction*, ISARC, Eindhoven, The Netherlands.

Sadeghpour, F., Moselhi, O., and Alkass, S. (2002). "Dynamic planning for site layout." *Proc., 30th Annual Conf. of Canadian Society of Civil Engineering*, CSCE, Montreal.

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." *J. Constr. Eng. Manage.*, 128(3), 220–231.

Tam, C. M., and Tong, T. K. L. (2003). "GA-ANN model for optimizing the locations of tower crane and supply points for high-rise public housing construction." *Constr. Manage. Econom.*, 21(3), 257–266.

Tommelein, I. D., and Zouein, P. P. (1993a). "Interactive dynamic layout planning." *J. Constr. Eng. Manage.*, 119(2), 266–287.

Tommelein, I. D., and Zouein, P. P. (1993b). "Space scheduling for construction progress planning and control." *Proc., 10th Int. Symp. on Automation and Robotics in Construction*, ISARC, Houston, Tex., 415–422.

Tommelein, I. D., Levitt, R. E., and Hayes-Roth, B. (1992a). "Site-layout modeling: How can artificial intelligence help?" *J. Constr. Eng. Manage.*, 118(3), 594–611.

Tommelein, I. D., Levitt, R. E., and Hayes-Roth, B. (1992b). "SightPlan model for site layout" *J. Constr. Eng. Manage.*, 118(4), 749–766.

Tommelein, I. D., Levitt, R. E., Hayes-Roth, B., and Confery, T. (1991). "SightPlan experiments: Alternate strategies for site layout design." *J. Comput. Civ. Eng.*, 5(1), 42–63.

Yeh, I. C. (1995). "Construction-site layout using annealed neural network." *J. Comput. Civ. Eng.*, 9(3), 201–208.

Zamanian, M. K. (1992). "Modeling and communicating spatial and functional information about constructed facilities." PhD dissertation, Engineering Design Research Center, Carnegie Mellon Univ., Pittsburgh.

Zouein, P. P. (1996). "MoveSchedule: A planning tool for scheduling space use on construction sites." PhD dissertation, Univ. of Michigan, Ann Arbor, Mich.

Zouein, P. P., Harmanani, H., and Hajar, A. (2002): "Genetic algorithm for solving site layout problem with unequal-size and constrained facilities." *J. Comput. Civ. Eng.*, 16(2), 143–151.

J. Constr. Eng. Manage., 2006, 132(2): 143-151