Improvement Algorithm for Limited Space Scheduling

By P. P. Zouein,[1] and I. D. Tommelein,[2] Associate Member, ASCE

Abstract: Selecting construction methods, scheduling activities, and planning the use of site space are key to constructing a project efficiently. Site layout and activity scheduling have been tackled as independent problems. Their interdependence is often ignored at the planning stage and may be dealt with—if at all—when construction is underway. Problems that may have had easy solutions if dealt with earlier, may then be expensive to remedy. This paper addresses the combined problem termed ''space scheduling'' and presents an algorithmic time-space trade-off model for adjusting activity durations and start dates to decrease the need for space over congested time periods. The model characterizes resource space requirements over time and establishes a time-space relationship for each activity in the schedule, based on alternative resource levels. An example illustrates the presented algorithm that generates a feasible space schedule.

INTRODUCTION

Selecting construction methods, scheduling activities, and planning the use of site space over time are key to constructing a project efficiently. Site layout and activity scheduling have been tackled as independent problems. Their interdependence is often ignored at the planning stage and may be dealt with —if at all—when construction is underway. Problems identified at that time are expensive to remedy; they may have been easy to resolve or even avoided altogether by better advance planning.

The combined problem of space planning and activity scheduling, known as ''space scheduling'' (Tommelein and Zouein 1993), is the subject of this paper. It involves scheduling activities subject to space availability constraints while minimizing the project duration. Space scheduling can be viewed as a 2 or 3D limited resource scheduling problem where the limited resource is space.

Fig. 1 illustrates space scheduling. Temporal project data is represented by a schedule of four activities, and associated resources represented by the patterned rectangles inside the activity boxes. This data is used in constructing site layouts over time (dynamic layout). The start and finish dates of the activities mark time periods during which specific sets of resources are needed on site. For example, in the time period 0–4, two activities take place. They require lumber, backfill material, and a crane (represented by the dotted, cross-hatched, and solid black rectangles, respectively) for their performance. These resources require space on-site, equal to the areas of the rectangles representing them. Constructing the layout for time period 0–4 then consists of finding feasible and economical positions for them. A layout of these resources (layout 0–4) is shown in the lower right-hand corner of Fig. 1, which also includes layouts for subsequent time periods.

Fig. 1 shows that construction of the dynamic layout helped identify a spatial conflict in time period 10–12. While positioning the resources needed during that time period, it became clear that the concrete batch plant cannot be on-site without overlapping other resources needed simultaneously. This spatial conflict can be solved by relocating other resources, delaying the activity requiring the use of the concrete batch plant, or using ready-mix concrete instead. These changes may affect the schedule as is illustrated by the arrows in Fig. 1.

[1]Asst. Prof., Industrial Engrg. Dept., Lebanese American Univ., Byblos, Lebanon. E-mail: pzouein@lau.edu.lb

[2]Assoc. Prof., Civ. and Envir. Engrg. Dept., Univ. of California, Berkeley, CA 94720-1712. E-mail: tommelein@ce.berkeley.edu

Note. Discussion open until September 1, 2001. To extend the closing date one month, a written request must be filed with the ASCE Manager of Journals. The manuscript for this paper was submitted for review and possible publication on December 20, 1999; revised August 22, 2000. This paper is part of the ***Journal of Construction Engineering and Management***, Vol. 127, No. 2, March/April, 2001. ©ASCE, ISSN 0733-9634/01/0002-0116–0124/$8.00 + $.50 per page. Paper No. 22145.

Thus Fig. 1 illustrates the interdependence between activity scheduling and site layout and the importance of considering the combined, space scheduling problem to generate feasible plans. This paper presents an improvement algorithm to solve this problem.

SPACE SCHEDULING MODELS

Space scheduling has received more attention in recent years, because schedule compression—now a common project requirement—leads to increased spatial interference among resources on-site. Rad (1980) and Mawdesley et al. (1988) presented early models that acknowledged the need for assessing site space availability while scheduling resources. Smith (1987) and Tommelein et al. (1992) quantified the dynamics of work space required by construction activities either surrounding or inside a facility being built. Others assessed space availability and needs while scheduling activities, either with user interaction (Tommelein et al. 1993; Riley 1994; Akinci et al. 1998) or using a fully automated approach (Morad and Beliveau 1991; Thabet and Beliveau 1994, 1997; Zouein and Tommelein 1994, 1999, Zouein 1995). However, these fully automated models, focused on space needs of activities inside buildings, (e.g., to store, fabricate, and assemble materials or building components just prior to or while performing work).

Space dynamics inside buildings are different from those in staging and laydown areas used for temporary storage of materials and equipment on-site. This paper focuses on the latter. These differences impose other modeling needs. For example, the fast space turnover inside the building may warrant more continuous and detailed monitoring of changes to check for interference, among resources. In contrast, the long term use of laydown space by resources awaiting relocation or use by production activities can tolerate discretization of time. Changes in space availability and needs must be checked only at selected points in time. Also, distances traveled by resources inside the building are smaller than distances traveled by resources in-between staging or work areas on a construction site. This may explain why costs related to travel distances (e.g., transportation and relocation costs) were ignored by most indoor space scheduling models, although they can be substantial on large sites and must certainly be considered in outdoor layout planning.

This paper focuses on solving spatial conflicts that are identified during space scheduling rather than solving the dynamic layout problem. Space scheduling allows for schedule changes whereas dynamic layout planning considers the schedule to be a given. Tommelein and Zouein (1993), Zouein (1995), and Zouein and Tommelein (1999) already have presented user-interactive as well as fully-automated methods for constructing dynamic layouts.

This paper presents an algorithmic approach for scheduling space. A 2D representation characterizes resource, and con-

J. Constr. Eng. Manage., 2001, 127(2): 116-124

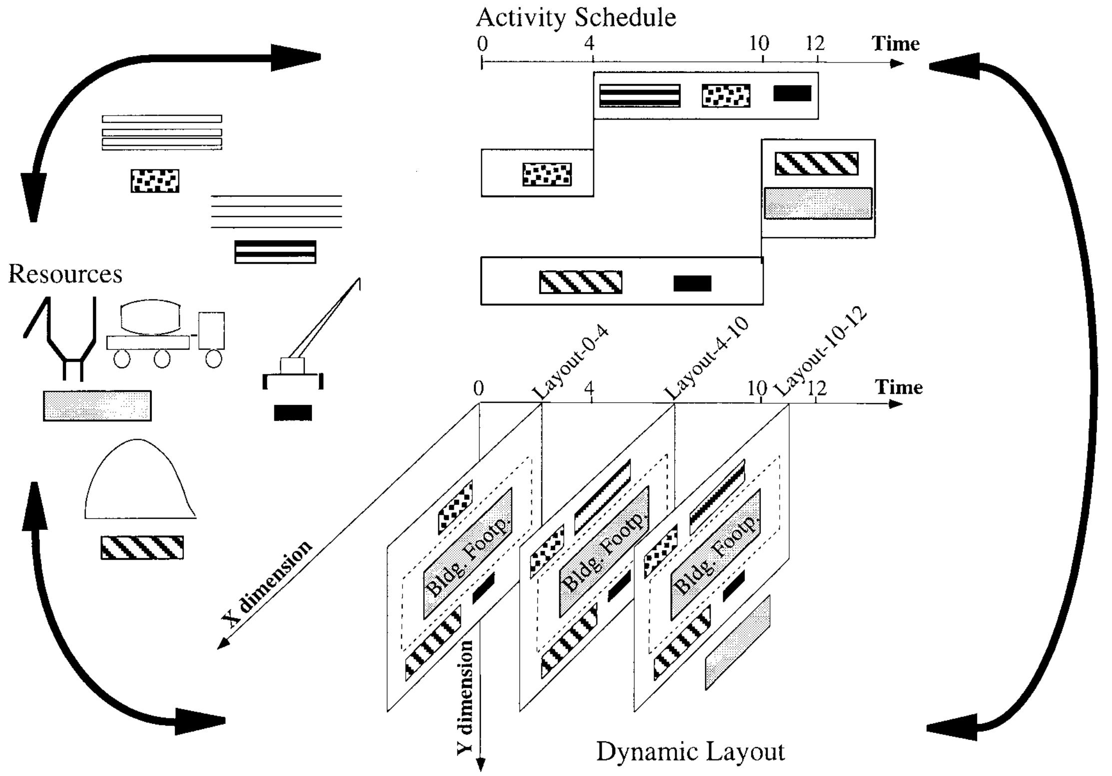

FIG. 1. Space Scheduling Process

sequently activity, space requirements over time. Improvement heuristics adjust an initial construction schedule to comply with site space limitations. Various time-space tradeoff strategies resolve spatial conflicts detected in the process of constructing the dynamic layout. The result is a feasible space schedule.

TIME-SPACE RELATIONSHIPS

Resource Area Profiles

A key issue in space scheduling is recognizing that space needs are dynamic, not only because resources are associated with activities but also because their space needs change as activities progress. By definition, dependent resources' space needs vary with the duration of the activity(ies); they are associated with and are present on-site for a time period corresponding to that(those) activity(ies)' start and finish time(s). In contrast, independent resources' space needs are constant throughout their presence on-site for a user-defined time period. Variations over time of dependent resources' space needs are modeled using three alternative area profiles: Profile A, B, and C. Space needs of independent resources are modeled using Profile D. Profiles are used to infer the area required by a resource at any one time during problem solving.

Profile A: Decreasing Space Need as Construction Progresses

Profile A models a resource whose space requirement decreases as the activity progresses. This is typical when the resource shortage penalty is high, large one-time deliveries are cost effective, and adequate means are available to accommodate the resource early. From a site space management standpoint, Profile A may not be desirable as a lot of space is tied up early. For example, Profile A can model construction materials (e.g., sand or gravel) that get consumed when the corresponding activity (e.g., an excavation or backfill activity) progresses. Fig. 2(*a*, left) is an example scenario. The total

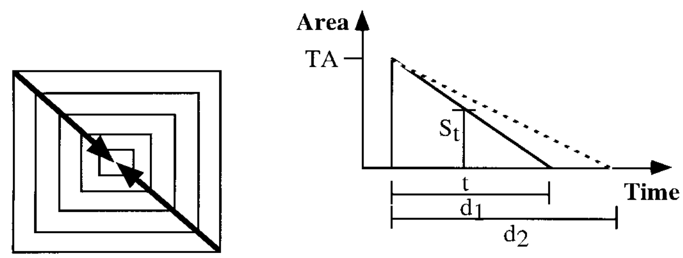

FIG. 2(a). Example Scenario (left), Profile A Model (right)

resource is brought to the site all at once, when the resource is first needed, and the stock is depleted as the activity progresses. The area the stock occupies on-site, therefore, shrinks over time, as shown by the arrows in Fig. 2(a, left).

Fig. 2(a, right) models the area of a Profile A resource. The total area (TA) required to accommodate the total resource is reserved on-site when the associated activity starts. The area S_t occupied by the resource at any one time t decreases linearly over the activity duration d_l. The dashed line in Fig. 2(a, right) shows that when the activity takes longer to complete (duration d_2), the resource consumption rate is slower; at any point in time, the resource will occupy more space on-site.

In the space scheduling algorithm, TA and a fixed length-to-width (L/W) ratio (defining the shape of each resource) must be defined by the user. The actual dimensions of the resource at any one time are then computed using L/W and the value of the area derived from the area profile at that time.

Profile B: Seemingly Constant Space Need as Construction Progresses

Profile B models a resource whose area requirement is either constant or fluctuates between a minimum and a maximum level as the activity progresses. It applies when space on-site is limited, and multiple deliveries are made. For example, Profile B can model construction materials (e.g., bricks or pipe) that are replenished on a regular basis during the correspond-

J. Constr. Eng. Manage., 2001, 127(2): 116-124

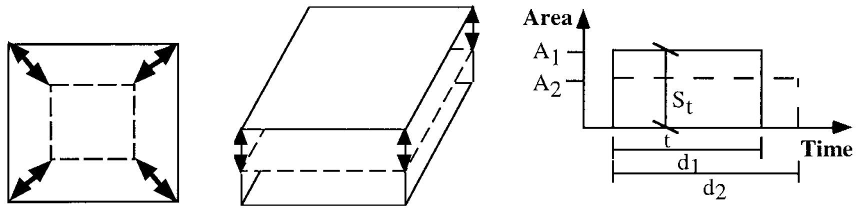

FIG. 2(b). Example Scenarios (Left and Middle), Profile B Model (Right)

ing activity, or that can be stacked so that they occupy a constant space on-site (e.g., roof trusses or plywood). Fig. 2(b) shows two example scenarios. In Fig. 2(b, left), the area occupied by the resource shrinks and expands as the resource is consumed and replenished over time. In Fig. 2(b, middle), the resource is stacked so that the occupied area remains constant, although the height varies over time.

Fig. 2(b, right) models the area of a Profile B resource. A constant area A_1 is reserved on-site to accommodate the resource at its maximum space requirement for the duration d_1 of the activity. When the activity is scheduled to progress at a slower pace (duration d_2), the area S_t required at any one time t will be smaller as indicated by A_2 and the dashed line in Fig. 2(b, right).

A_i, and, therefore S_t, are user-defined for each activity duration d_i (where i = 1, 2, etc.). The actual dimensions of the resource are computed using a user-defined L/W ratio and remain constant over the activity duration.

Profile C: Constant Space Need as a Function of Construction Progress

Profile C models a resource requiring a constant amount of space over time. The resource may depend on one, or several, activities, but it is neither consumed nor generated by them. For example, Profile C can model a piece of construction equipment that is assigned to more than one activity and remains idle between activities. The space required by the equipment is constant, as defined by its footprint, plus any additional area for maneuvering.

Fig. 2c models the area of a Profile C resource. A rectangle of length L and width W, input by the user, models the shape of the area required to accommodate the resource over the time interval TB, during which it remains on-site. By assumption, TB extends from the start of the first to the end of the last activity requiring its use.

Profile D: Constant Space Need Not a Function of Construction Progress

Profile D models an independent resource whose space requirement is constant for the duration it is present on-site. For example, Profile D can model facilities that support construction operations (such as trailers, parking lots, or obstacles on-site, such as trees or existing buildings), or some construction equipment needed to support construction operations in general (such as a tower crane).

Fig. 2(c) also models the area of a Profile D resource. A rectangle of length L and width W reflects the shape of the space required to accommodate the resource over a time interval TB during which the resource is on site; L, W, and TB for each resource are input by the user.

Discretizing Space Requirements

Changes in space requirements take place at discrete points in time. These points in time delimit what are termed primary time frames (PTFs). PTFs slice the project duration into dis-

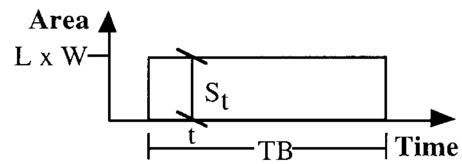

FIG. 2(c). Profile C and Profile D Models

crete time intervals, based on the arrival and departure of resources to and from the modeled site space. The time when a resource arrives on-site marks a new demand for space. This demand must be satisfied by finding a position for the resource, given its space needs, plus the current and possibly future constraints on space use. The time when a resource is removed from site marks an opportunity for other resources to occupy the freed space. These arrival and departure times are inferred from the schedule.

Each PTF defines the duration of a layout. Each resource in a layout presumably occupies space corresponding to the maximum area needed by it during that time frame (as inferred from the resource's area profile). For example, if a Profile A resource is needed in PTF-e-f, then an area equal to S_e (where $S_e \leq$ TA) is needed to accommodate the resource for the time interval e to f. For all other profiles, the required area is constant and hence no assumption on the timing of the space need is necessary.

ACTIVITY RESOURCE LEVELS

An activity typically requires different resources in various quantities, or of alternative type, for its performance. Resource levels model such alternative durations and area requirements. In the presented algorithm, a minimum of one and maximum of three (minimum, normal, and maximum) resource levels can be user-defined for each activity to describe alternative durations (maximum [d_{max}], normal [d_{norm}], and minimum [d_{min}]; note inverse order: d_{max} > d_{norm} d_{min}) and area requirements. Each resource level consists of a combination of Profile A, B, or C resources.

The total area required for the activity is the sum of the individual resources' area requirements at the assigned resource level. Lowering the activity's resource level can decrease or increase its total area requirement, depending on the type of profiles used.

PROBLEM-SOLVING METHOD

The problem-solving method tackles one PTF at a time and alternates between (1) constructing a layout for that PTF; and (2) adjusting the schedule to resolve any space conflicts. Schedule adjustments require as user input: (1) construction activities, durations, and precedence relationships (i.e., the activity network); (2) the arrival and departure times of independent resources; and (3) the resource levels for each activity. Layout construction may be done manually (Tommelein et al. 1993; Tommelein and Zouein 1993) or using heuristic optimization (Zouein 1995; Zouein and Tommelein 1999). The

J. Constr. Eng. Manage., 2001, 127(2): 116-124

latter may require additional user input regarding: (4) the level of interaction (termed proximity weight) between any two resources; (to reflect the amount of material flow or transportation cost); (5) constraints on the relative positions of resources in a given PTF (referred to as hard constraints); (6) penalty or cost of moving a resource from one position to another on-site (termed relocation weight); and (7) the positions of resources with user-defined fixed positions on-site (termed static resources) if applicable.

The objective is to develop a schedule of activities with the shortest project duration that complies with site space limitations. This is achieved by constructing a feasible dynamic layout that minimizes transportation and relocation costs of resources. The feasibility of the dynamic layout is established by satisfying the space requirements of all resources, ensuring that all hard constraints among them are met, and not relocating resources with prohibitive relocation cost (termed stationary).

The method starts when the unconstrained schedule corresponds to the shortest project duration (by setting each activity to be performed at its shortest duration) and then lengthens the project duration, as needed, during dynamic layout construction to decrease the total space required in problematic time frames. Starting with the shortest duration schedule is desirable, because this schedule corresponds to the highest demand for space and is likely to lead to the largest number of spatial conflicts.

During problem solving, the following assumptions hold: (1) all resources other than space exists in unlimited quantities; (2) a resource cannot have different area profiles in different PTFs; (3) activities cannot be interrupted and, once an activity has started, no changes can be made to its start data or resource level; and (4) activities have finish-to-start precedence relationships and should not have a mandatory start or finish date. These assumptions are commonly made in algorithmic approaches to solve limited resource scheduling problems.

Space Scheduling Algorithm

The space scheduling algorithm (SSA) iterates through the following steps until layouts for all PTFs have been successfully constructed and no spatial conflicts remain:

1. Compute the unconstrained schedule corresponding to the shortest project duration and identify PTFs based on the schedule and the timing of independent resources.
2. Construct the layout of PTFs, one at a time, and in sequential order.
3. Declare a spatial conflict in the PTF if the method used for layout construction fails to find positions for all resources in the selected PTF (say PTF-e-f) that satisfy user requirements or predefined constraints between resources. Otherwise, the method succeeded in finding a feasible layout for that PTF and moves to step 7.
4. If a spatial conflict is declared, a time-space trade-off strategy (Appendices I and II) is selected or a user-labeled resource is removed from the PTF to reduce the total area requirement in it at minimal increase in project duration. If no conflict is detected, then go to Step 6.
5. Recompute start and finish dates of activities based on the time-staff tradeoff strategy selected and reidentify the resulting PTFs.
6. Start anew with constructing a layout for PTF-e-f or for a replacement PTF-e-g, because changes to the schedule may have affected the finish date of the first PTF, the resources present in it, and constraints on their positions. Go to Step 3.
7. If the layouts for all PTFs are successfully constructed, the current schedule is final and the space scheduling procedure terminates; otherwise some PTFs remain to be laid out, so loop back to Step 2.

Note that starting with an early-start and shortest-duration schedule de facto limits the possible changes to either lengthening or delaying activities, thus forcing the changes to be in a forward pass only.

Time-Space Tradeoff Strategies

When a spatial conflict is detected in the process of constructing the layout of PFT-e-f, SSA will select a strategy from the set of strategies described in Appendix I. These strategies may either delay the start date of an activity or lower its resource level to reduce the total area requirement in the problematic time frame. Strategy selection (Appendix II) can be done in one of two modes: a fully-automated mode; or a user-driven mode where the user labels some dependent resource to be removed from PTF-e-f. In each mode, the space scheduling algorithm prioritizes the strategies to select the one that reduces the total area requirement in PTF-e-f at a minimum increase in project duration. Reducing the total area requirement will increase the amount of available site space and may help solve the spatial conflict in this PTF.

EXAMPLE APPLICATION

Input

This example illustrates the use of the space scheduling algorithm to assess the feasibility of the schedule for a concrete foundation project and then to adjust this schedule to comply with site space limitations. This example is small, due to page length limitations, but is clearly illustrates the method. The activity network shown in Fig. 3(a) describes construction of the north and east foundation walls of a building. The trenching and excavation work preceding the foundation walls' construction are completed and are not shown in the network.

Is Fig. 4(a) site dimensions are 22×12 units. No Dimensions are given to these units as they are irrelevant to the problem-solving method. The north trench is 2 units wide and 20 units long and runs along the northern site limit. The west trench is 2 units wide and 12 units long and runs along the west side. The east side of the site is fenced. The south side is only partially fenced with a reserved area of 4 by 3 located at 12 units from the southeast corner of the site, which serves as site access during this phase of construction.

Trenches, fence lines, and site access are modeled as Profile D resources—on-site for the duration of construction of this foundation wall. They are stationary and have a fixed, user-defined position on-site [Fig. 4(a)]. Since the duration of this construction phase is not known at the time the independent resources are input, the user can assign an arbitrarily large number to TB to guarantee that the resource will be on-site for the entire project duration. The resource levels for each activity and data defining the resources in each level are input by the user (Tables 1 and 2). For the sake of brevity, only two resource levels are defined per activity.

Resources with fixed shape and dimensions (e.g., the pipe layer and the concrete batch plant) and resources used by more than one activity that remain idle on-site in-between uses (e.g., the lumber shop and the welding shop) are modeled using Profile C. Other dependent resources that get consumed in the construction process (e.g., aggregates, cement, rebar, pipe) are modeled using Profile A and Profile B, depending on whether the total quantity of the resource is brought to the site in one, or multiple, deliveries.

The relocation of the lumber shop and the concrete plant are restricted in this example. These resources may be relo-

J. Constr. Eng. Manage., 2001, 127(2): 116-124

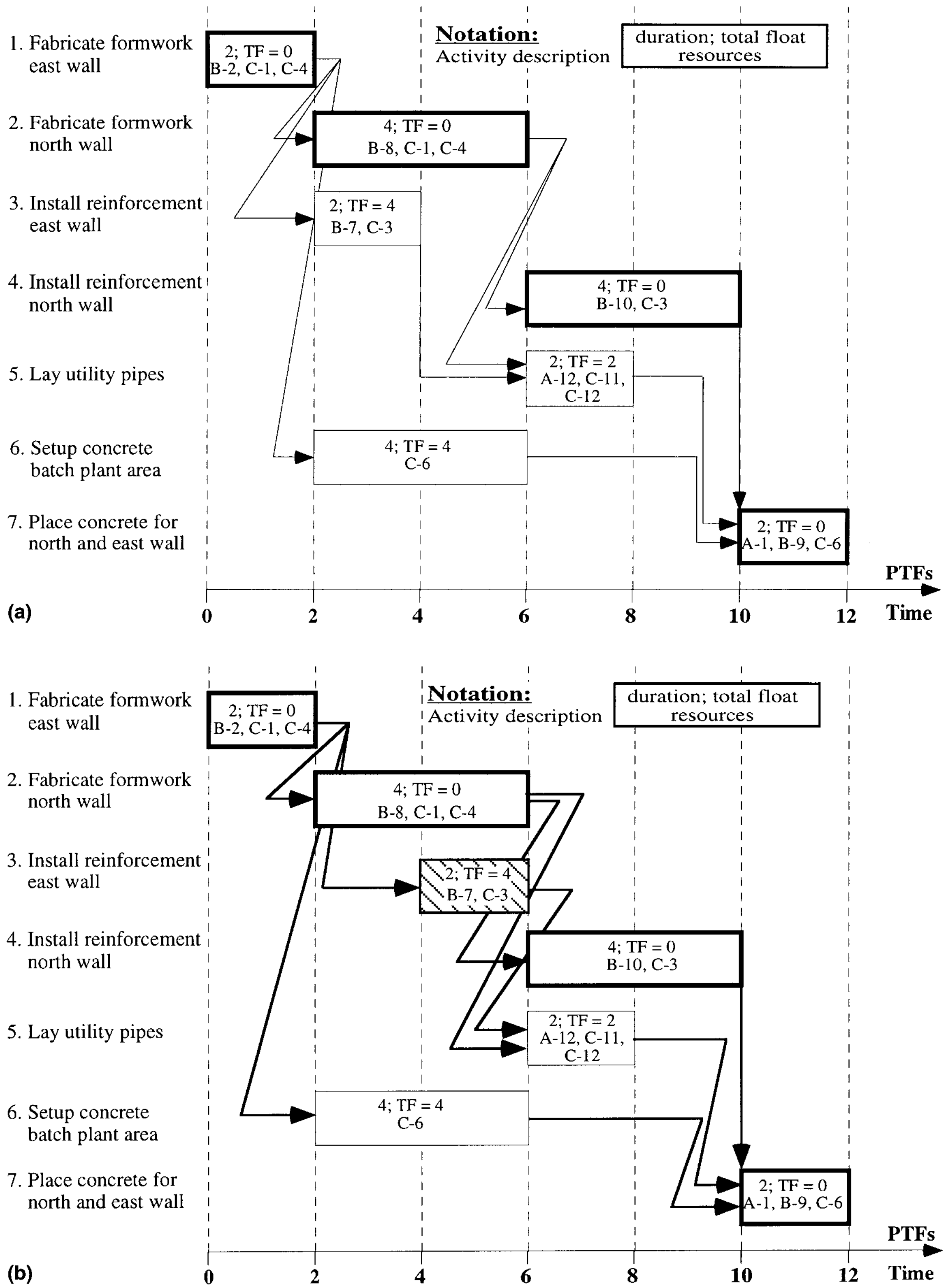

FIG. 3. (a) Unconstrained Schedule; (b) Updated Schedule and PTFs

catable, but they are modeled here as stationary (i.e., they cannot be relocated in different PTFs) to illustrate how positioning stationary resources may constrain the positions of other resources in subsequent layouts in a forward-pass approach.

Computations

Initially, SSA schedules all activities to be performed at their shortest duration, computes the corresponding unconstrained schedule, and identifies the PTFs [Fig. 3(a)]. The feasibility of the unconstrained schedule is assessed by constructing the dynamic layout; DLA constructs the layouts of the individual PTFs in chronological order. Only the construction layout of PTF-0-2 and PTF-2-4 are presented in this example to illustrate how SSA words and solves spatial conflicts in the course of dynamic layout construction.

Layout Construction of PTF-0-2

The resources that need positioning in PTF-0-2 are lumber for the east wall (B-2), the lumber shop (C-1), and the tool

J. Constr. Eng. Manage., 2001, 127(2): 116-124

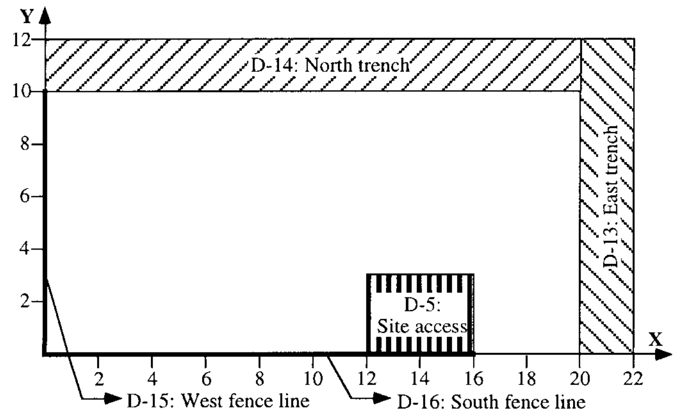

FIG. 4(a). Site Layout with Long-Term Stationary Resources for Example Project

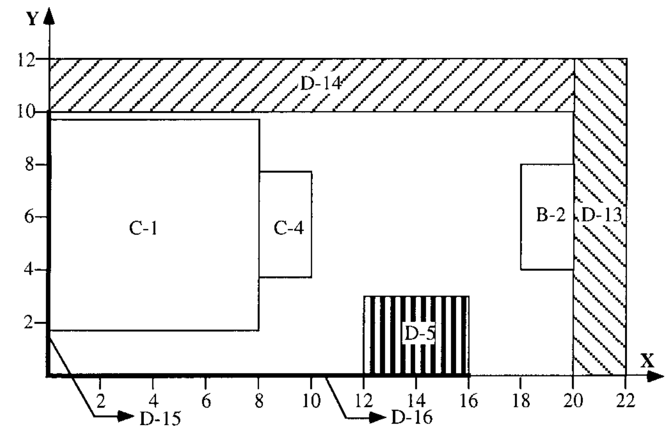

FIG. 4(b). Final Layout for PTF-0-2

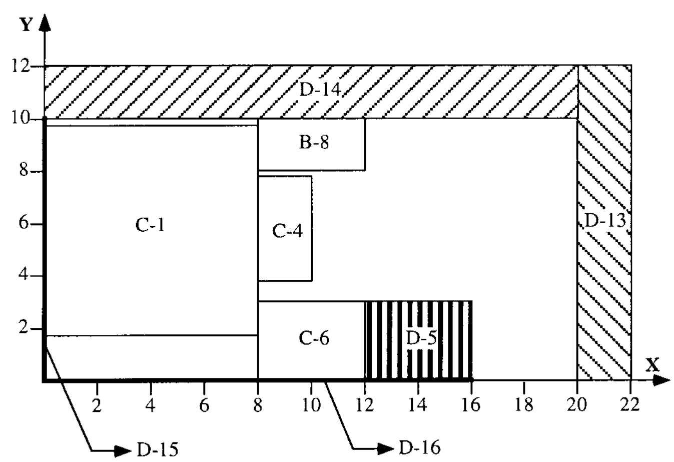

FIG. 4(c). Final Layout for PTF-2-4

trailer (C-4). Other resources that occupy space on-site are the static resources D-5, D-13, D-14, D-15, and D-16. The layout construction method uses heuristic optimization to derive positions of resources, which minimize the sum of the weighted distances between them and their relocation costs (Zouein and Tommelein 1999). This method requires additional user input regarding proximity weights and hard constraints between resources in each PTF. Table 3 shows these for PTF-0-2.

The dimensions of B-2 are computed using its area requirement at time 0 (i.e., 8 from Table 1) and its L/W ratio (i.e., 2 from Table 2) and are L = 4 and W = 2.

Layout construction results in the following positions for the resources in PTF-0-2: X1 = 4, Y1 = 5.95 @ 0°, X2 = 19, Y2 = 6 @ 90°; X4 = 9, Y4 = 5.95 @ 90°. The final layout for PTF-0-2 is shown in Fig. 4b.

Layout Construction of PTF-2-4

The resources that need positioning in PTF-2-4 are the stationary and relocatable resources from the layout of PTF-0-2, C-1, C-4, and the new resources B-7, B-8, C-3, and C-6. The independent and static resources D-5, D-13, D-14, D-15, D-16 remain at their known positions.

The user is asked to input proximity weights and hard constraints between pairs of resources in PTF-2-4 (Table 4). The hard constraints concern the location of: (1) the welding shop (C-3), which should be at least 8 units away from any combustible material storage and 7 units away from the concrete plant (C-6) for safety reasons; and (2) the concrete plant (C-6) which should be along the southern fence line (D-16) for easy access.

First, the dimensions and area requirements of B-7 and B-8 in PTF-2-4 are determined using their area requirement at time 2 and their respective L/W ratio (taken from Tables 1 and 2, respectively). The area required by B-7 is 8, its L/W is 2; and its dimensions are therefore L = 4 and W = 2. Similarly, the dimensions of B-8 are found to be L = 4 and W = 2. The dimensions of all Profile C and Profile D resources are as input.

Given this situation, the layout construction method fails to find a feasible position for C-6 and C-3 that satisfies the hard constraints on their positions relative to C-1's current position. Hence, PTF-2-4 layout construction halts and the spatial conflict, detected, should be solved.

Solving Spatial Conflict in PTF-2-4

Assume Mode 1 is selected to solve the conflict. Accordingly, SSA looks for a strategy-activity combination to decrease the total area requirement in PTF-2-4 at minimum increase in project duration.

Under Strategy A, an activity is delayed to start at time 4. The candidates are activities 2, 3, and 6 with total float values of 0, 4, and 4, respectively. If delayed, their remaining-total-float values are −2 (i.e., 0 [the total float of activity 2] − 2 [the delay]), 2, and −2 respectively. Activities 3 and 6 tie as candidate activities, because they have the same remaining-total-float value. Strategy A breaks the tie by selecting activity 3 because delaying it would bring a larger reduction in total area requirement in PTF-2-4 (i.e., an area of 15.84 if activity 3 is delayed versus 12 if activity 6 is delayed from Table 1).

Under Strategy B, an activity's resource level is lowered. The candidates are activities 2 and 3 because both start at time 2 and their area requirement is smaller if performed at a lower resource level (Table 1). Although activity 6 also starts at time 2, it is not considered, because it cannot be performed at a lower resource level (Table 1). The decrease in the area requirement of activities 2 and 3 are 2 (i.e., 80 − 78) and 4 (i.e., 15.84 − 11.84), respectively (Table 1); the increase-in-activity-duration are 2 (i.e., 6 − 4) and 2 (i.e., 4 − 2), respectively (Table 1). Hence the remaining-total-float of activity 2 is 0 (its total float) minus 2 (its increase-in-activity-duration) that equals −2. Similarly, the remaining-total-float of activity 3 is 4 minus 2 which equals 2. Strategy B returns activity 3, because it has the largest value of remaining-total-float and lowering its resource level will reduce the total area requirement in PTF-2-4 by 4.

SSA selects the strategy-activity combination with the highest value of remaining-total-float. Both Strategies A and B tie with a value of 2. SSA breaks the tie by selecting the one that causes the largest decrease in the total area requirement in PTF-2-4, which is Strategy A and activity 3. Therefore, activ-

J. Constr. Eng. Manage., 2001, 127(2): 116-124

TABLE 1. Activities and Resource Levels for Example Project

Activity number (1)	Description (2)	Duration (3)	Resource Level: Level (4)	Resource Level: [Type, area] (5)	Area at start (6)
1	Fabricate formwork east wall	2	Normal	[B-2, 8] [C-1], [C-4]	80
		4	Minimum	[B-2, 6], [C-1], [C-4]	78
2	Fabricate formwork north wall	4	Normal	[B-8, 8], [C-1], [C-4]	80
		6	Minimum	[B-8, 6], [C-1], [C-4]	78
3	Install reinforcement east wall	2	Normal	[B-7, 8], [C-3]	15.84
		4	Minimum	[B-7, 4], [C-3]	11.84
4	Install reinforcement north wall	4	Normal	[B-10, 8], [C-3]	15.84
		5	Minimum	[B-10, 6], [C-3]	13.84
5	Lay utility pipes	2	Normal	[A-12, 12], [C-11 C-12]	24
		4	Minimum	[A-12, 12], [C-11]	18
6	Setup concrete batch plant area	4	Normal	[C-6]	12
7	Place concrete for north and east wall	2	Normal	[A-1,6], [B-9, 9], [C-6]	27
		1	Minimum	[A-1, 6], [C-9, 4], [C-6]	22

TABLE 2. Dependent Resource Data for Example Project

Profile (1)	Description (2)	L/W ratio (3)	L × W (4)	Relocation weight (5)	Fixed position (6)
A-1	Aggregates	1.5	—	100	—
A-12	Utility pipes	2	—	100	—
B-2	Lumber east	2	—	25	—
B-7	Rebar east	2	—	25	—
B-8	Lumber north	2	—	25	—
B-9	Cement bags	1	—	50	—
B-10	Rebar north	2	—	25	—
C-1	Lumber shop	—	8 × 8	Stationary	—
C-3	Welding shop	—	2.8 × 2.8	0	—
C-4	Tools trailer	—	4 × 2	100	—
C-6	Batch plant	—	4 × 3	Stationary	—
C-11	Pipelayer 1	—	3 × 2	0	—
C-12	Pipelayer 2	—	3 × 2	0	—

TABLE 3. Proximity Weights and Hard Constraints in PTF-0-2

Proximity Weights			Hard Constraints			
Resource 1 (1)	Resource 2 (2)	Weight (3)	Resource 1 (4)	Resource 2 (5)	Constraint type (6)	Value (7)
B-2	C-1	50	C-1	D-15	max Dx	0
B-2	D-13	200	—	—	—	—
C-1	C-4	100	—	—	—	—

TABLE 4. Proximity Weights and Hard Constraints in PTF-2-4

Proximity Weights			Hard Constraints			
Resource 1 (1)	Resource 2 (2)	Weight (3)	Resource 1 (4)	Resource 2 (5)	Constraint type (6)	Value (7)
B-7	C-3	100	C-3	C-1	min D_x	8
B-7	D-13	200	C-3	C-6	min D_x	7
B-8	C-1	50	C-6	D-16	max D_y	0
B-8	D-14	200	—	—	—	—
C-1	C-4	100	—	—	—	—

ity 3 is delayed to start at time 4, and the unconstrained schedule and the PTFs are updated to reflect this change [Fig. 3(b)].

Layout Construction of PTF-2-4 after Schedule Modification

Construction of the layout for PTF-2-4 is started anew, now including only B-8, C-1, C-4, and C-6, in addition to the static resources. The user may make changes to the proximity weights and hard constraints in PTF-2-4. Assume all remain as shown in Table 4, except the proximity weights and hard constraints involving C-3 and B-7 are ignored, because these resources no longer exist in this PTF. The area of B-8 is as computed earlier for time 2.

By reapplying the layout construction method to the new set of resources and constraints, the algorithm finds the following feasible positions for the resources in PTF-2-4; X_6 = 10 and Y_6 = 1.5 @ 0°; X_8 = 10 and Y_8 = 9 @ 0°; and X_4 = 9 and Y_4 = 5.95 @ 90°. The latter position happens to be C-4's position in PTF-0-2, so this resource need not be relocated.

Output

The model yields feasible site layouts for the time interval 0–2 and 2–4 and an adjusted activity schedule for this time period. The final layout solutions for PTF-0-2 and PTF-2-4 are as shown in Fig. 4b and Fig. 4c, respectively. The adjusted schedule is as shown in Fig. 3 (bottom). For this schedule to be final, feasible layouts for the remaining PTFs should be constructed.

METHOD PRACTICALITY AND LIMITATIONS

This model characterizes a variety of resources whose area requirements either decrease or stay constant for the duration of an activity, or are constant and independent of the duration of an activity. Actual area profiles of resources may vary significantly, depending on the nature of the resource, the project, the nature of the site, and the company's procurement practices. The proposed profiles are not exhaustive but can be used as a conservative approximation of more exact profiles. For example, the area requirement of a resource that builds up onsite as the corresponding activity progresses (e.g., earth during excavation or forms during stripping of formwork) can be conservatively modeled using Profile B or C where a constant area is reserved to accommodate the resource at some approximate space requirement.

The model allows the user to define resources that are shared by multiple activities, which may nor may not take place concurrently. Resource levels are set arbitrarily to 3 but SSA can handle any number provided that activities initially are scheduled to be performed at their shortest duration.

The space scheduling algorithm uses a forward pass approach for modifying an initial schedule to comply with spatial limitations on-site. The method starts with the shortest-duration schedule and modifies it by lengthening or delaying activities. A forward-pass approach is meaningful, because it can address spatial conflicts that were not anticipated during planning but arose during construction. Interrupting on-going activities may then not be practical or aid in solving the conflict, because their corresponding resources are likely to remain onsite and, therefore, continue to claim space.

The algorithm can be extended by mechanisms that allow

J. Constr. Eng. Manage., 2001, 127(2): 116-124

SSA to undo the actions of previously applied strategies, if a spatial conflict persists in the problematic time frame. As it stands, SSA cannot judge whether the applied strategy was ineffective (i.e., did not contribute to solving the conflict in the problematic time frame) or that a combination of strategies is needed to solve the conflict. SSA can most benefit from augmentation with mechanisms to reason about the cause of a conflict so that it can intelligently select strategies to undo. To some extent, Mode 2 was intended to address this weakness by allowing the user to suggest ways for solving the conflict. Mode 2, however, should be extended to benefit from strategies that change the resource level of an activity, not just delay it or remove a selected resource and store it off-site.

CONCLUSIONS

This paper presents a time-space tradeoff algorithm that uses resource levels to describe alternative methods for performing an activity with different durations. Each level imposes another demand for space over an activity's duration. The paper presents a heuristic improvement algorithm that changes the activity schedule when insufficient space is available. The algorithm varies resource levels of activities or delays their start date to vary the demand for space on-site over problematic time frames while minimizing the increase in project duration.

This system addresses one aspect of space scheduling, namely, sequential layout construction with schedule adjustment. The user must decide on how to model space occupied by resources. The system presented here performs a forward pass only, and its solutions are, therefore, shortsighted. This reflects what often happens on-site, when space problems are not anticipated. However, the system makes it possible to conduct what/if analyses to prevent problems and optimize. In addition, the system's graphical display aids in visualizing the situation.

Space scheduling is a complex, though ubiquitous, problem. Finding solutions is difficult. The strategies presented here formalize alternative ways to resolve space conflicts, though they are not guaranteed to succeed at all times and their enumeration is not exhaustive.

APPENDIX I. TIME-SPACE TRADEOFF STRATEGIES

Strategy A: Delay Activity

Strategy A delays an activity that starts at the start of PTF-e-f (i.e., time e), to start at the end of PTF-e-f (i.e., time f). Assume:

- candidates = activities that start at time e
- delay = f − e
- remaining-total-float$_i$ = total float of activity i − delay
- current-area-requirement$_i$ = sum of area requirements of the resources in the current resource level of activity i at time e
- decrease-in-area-requirement$_i$ = current-area-requirement$_i$ − total area requirement of all Profile C resources of i that were brought to the site before e (i.e., Profile C resources associated with activities that started before time e).

This strategy selects from candidates the activity: (1) that has the maximum value of remaining-total-float; in case of a tie it selects the one (2) that has the largest value of decrease-in-area-requirement; and if still tied, it selects (3) one activity at random. This strategy may or may not increase the project duration depending on the values of the total float and remaining-total-float of the selected activity. If this strategy is selected, SSA resets the start date of the selected activity to be no earlier than time e and recomputes the CPM schedule and all PTFs, including and following the current one.

Strategy B: Lower Resource Level of Activity

Strategy B lowers the resource level of an activity that starts at the start of PTF-e-f and, therefore, lengthens its duration. Assume:

- candidates = activities starting at time e, whose total area requirement in PTF-e-f is lower if performed at a lower resource level
- increase-in-activity-duration$_i$ = duration of activity i at the current resource level − duration of activity i at the lower resource level
- remaining-total-float$_i$ = total float of activity i − increase-in-activity-duration$_i$
- current-area-requirement$_i$ = sum of area requirements of the resources in the current resource level of activity i at time e
- decrease-in-area-requirement$_i$ = current-area-requirement$_i$ − total area requirement of activity i if performed at a lower resource level

This strategy selects from candidates the one (1) that has the maximum value of remaining-total-float; in case of a tie, it selects the one (2) that has the largest value of decrease-in-area-requirement; and if still tied, it selects (3) one at random. This strategy may or may not increase the project duration, depending on the values of the total float and remaining-total-float of a candidate activity. If this strategy is selected, then SSA: (1) resets the duration of the selected activity to that of the lower resource level; (2) updates the resources assigned to it; and (3) recomputes the CPM schedule and all PTFs, including and following the current one.

Strategy C: Store Off-Site

Strategy C removes a Profile C resource that requires space on-site if it is idle during the time period e to f. It assumes that storage off-site is available. Resources modeled with any other profile exist on-site for the duration of their associated activity or for a user-defined time period and, hence, are considered in use in PTF-e-f. Before selecting this strategy, the user is asked to confirm that removal of the resource from site while it is idle is permissible. Selecting this strategy has no effect on the schedule and consequently on the PTFs.

APPENDIX II. STRATEGY SELECTION

Model 1: Fully Automated Mode

In Mode 1, the algorithm takes the problematic time frame PTF-e-f as input and selects either Strategy A or B. Either strategy may return an activity with a value for the remaining-total-float and a value for the expected decrease-in-area-requirement in PTF-e-f. The algorithm selects the strategy-activity combination with the largest value of remaining-total-float. In case of a tie, the activity with largest value of decrease-in-area-requirement is selected. If further tied, the algorithm selects one at random. The expected increase in project duration is equal to

- 0; if remaining-total-float of selected strategy ≥ 0
- 0 − remaining-total-float; if remaining-total-float of selected strategy < 0

SSA in Mode 1 will always find at least one strategy to apply, and thus will always return a modified schedule. Whether this schedule enables the spatial conflict to be solved

J. Constr. Eng. Manage., 2001, 127(2): 116-124

in PTF-e-f is determined by constructing the dynamic layout of PTF-e-f again. Several iterations may be needed before the layout of PTF-e-f is successfully constructed.

Mode 2: User Driven Mode

In Mode 2, the user labels a dependent resource R in the layout of PTF-e-f to be removed from the layout. Depending on R's profile, SSA selects a strategy that removes it from the layout at a minimum increase in project duration. If R is a Profile A or B resource, then SSA will delay the activity to which R is assigned to start at the end of PTF-e-f, provided the activity starts at time e. If R is a Profile C resource and it is idle in PTF-e-f, then Strategy C is applied, otherwise SSA will delay all activities to which R is assigned to start at the end of PTF-e-f, provided these activities start at time e. In all other cases (if R is assigned to activities that started in earlier PTFs), SSA fails to suggest a strategy since by assumption activities cannot be interrupted and only Mode 1 can be used.

APPENDIX III. REFERENCES

Akinci, B., Fischer, M., and Zabelle, T. (1998). ''Proactive approach for reducing non-value adding activities due to time-space conflicts.'' *Proc., 6th Annu. Conf. Int. Group for Lean Constr.*, IGLC-6, Guaruja, Brazil.

Mawdesley, M. J., Cullingford, G., and Haddadi, A. (1988). ''Site layout and resource scheduling—An approach to modelling movement around sites.'' *Proc., 5th Int. Symp. on Robotics in Constr.*, ISARC, Tokyo.

Morad, A. A., and Beliveau, Y. J. (1991). ''Knowledge-based planning system.'' *J. Constr. Engrg. and Mgmt.*, ASCE, 117(1), 1–12.

Rad, P. (1980). ''Analysis of working space congestion from scheduling data.'' *Am. Assn. Cost Engrs. Trans.*, F4.1–F4.5.

Riley, D. R. (1994). ''Modeling the space behavior of construction activities.'' PhD thesis, Arch. Engrg. Dept., Pennsylvania State Univ., University Park, Pa.

Smith, D. M. (1987). ''An investigation of the space constraint problem in construction planning.'' MS thesis, Civ. Engrg. Dept., Virginia Polytechnic Institute and State Univ., Blacksburg, Va.

Thabet, W. Y., and Beliveau, Y. J. (1994). ''Modeling work space to schedule repetitive floors in multistory buildings.'' *J. Constr. Engrg. and Mgmt.*, ASCE, 120(1), 96–116.

Thabet, W. Y., and Beliveau, Y. J. (1997). ''SCaRC: Space-constrained resource-constrained scheduling system.'' *J. Comp. in Civ. Engrg.*, ASCE, 11(1), 48–59.

Tommelein, I. D., Castillo, J. G., and Zouein, P. P. (1992). ''Space-time characterization for resource management on construction sites.'' *Proc., 8th Conf. Comp. in Civ. Engrg.*, ASCE, New York, 623–630.

Tommelein, I. D., and Zouein, P. P. (1993). ''Interactive dynamic layout planning.'' *J. Constr. Engrg. and Mgmt.*, ASCE, 119(2), 266–287.

Tommelein, I. D., Dzeng, R. J., and Zouein, P. P. (1993). ''Exchanging layout and schedule data in a real-time distributed environment.'' *Proc., 5th Int. Conf. Comp. in Civ. and Build. Engrg.*, ASCE, New York, 947–954.

Zouein, P. P., and Tommelein, I. D. (1994b). ''Time-space tradeoff strategies for space-schedule construction.'' *Proc., 1st Congr. Comp. in Civ. Engrg.*, ASCE, New York, 2, 1180–1187.

Zouein, P. P. (1995). ''Move schedule: A planning tool for scheduling space use on construction sites.'' PhD thesis, Civ. and Envir. Engrg. Dept., Univ. of Michigan, Ann Arbor, Mich.

Zouein, P. P., and Tommelein, I. D. (1999). ''Dynamic layout planning using a hybrid incremental solution method.'' *J. Constr. Engrg. and Mgmt.*, ASCE, 125(6), 400–408.

J. Constr. Eng. Manage., 2001, 127(2): 116-124