Auto Generating Construction Plan Alternatives Using A 5D Analyzer System

Mojtaba Taiebat, PhD Student, **KiHong Ku,** PhD
Virginia Polytechnic Institute and State University
Blacksburg, Virginia, USA

This paper discusses a framework to integrate 3D, cost, and schedule information, focusing on the generation of alternative construction methods. The objective is twofold: (1) to examine the state-of-the-art of current research and commercial developments in 5D modeling; and (2) to improve the concept of 5D integration by describing a data model framework. A wide range of research has been focusing on 4D CAD, virtual construction and prototyping. Despite the growing attention on Building Information Modeling by practitioners, the integration of multi-dimensional information of 3D, schedule, and cost remains limited. To address the limitations of previous research, this paper proposes a data model approach which integrates construction planning and methods knowledge via data attributes (i.e., production rate, crew size, work sequence). Integrating this data modeling approach with parametric and object-oriented modeling concepts, this approach can simulate alternative cost-duration scenarios based on owner, architect, and constructor criteria.

Key Words: BIM, Auto Schedule Generation, Object-Oriented Modeling, 4D, 5D

Introduction

The current trend towards globalization forces the AEC community to change its practical approach from a non-systematic, labor-intensive to a systematic one. The need for the paradigm shifted from ‘drawing’ to systematically ‘constructing’ for design practices is also recognized (Jin et al. 2007).

The critical path method (CPM) and bar charts have still been widely employed by project teams as a main tool to express the project schedules and coordinate the activities of members of project team (Huang et al. 2007). Some construction planners use the CPM method to integrate the product (i.e. what is to be done) with the process (i.e. how it is done) (Jongeling et al. 2007), but this leads to very detailed CPM schedules that are difficult to use and to update (Huber, 2003). As a result, detailed schedules are often not updated during a construction process and thereby lose their value as an instrument to plan and control workflow (Jongeling et al. 2007).

Location-based and activity-based scheduling are two other techniques currently being used in site planning and management. But Jongeling & Olofsson (2007) believe that neither location-based nor activity-based scheduling techniques provide users with insight in the spatial configuration of scheduled construction operations. 4D CAD models allow project planners to simulate and analyze what-if scenarios before commencing work execution on site (Mallasi et al. 2002). Planning supported by visual analyses of 4D CAD models is considered more useful and better than traditional planning (Fischer et al. 2004, Heesom et al. 2004).

Literature shows that a lot of efforts have been made for simulation, 4D modeling, and nD modeling to consider different aspects of the project before construction phase. In this paper some prototypes and commercial packages which go beyond 3D modeling are presented. Then a critique on the current state of the art shows the gap in this area, and the proposed system offers a way ahead for the future of construction planning.

Developed Prototypes

Not all simulation strategies suit the construction-oriented simulation well because of the uniqueness of construction operations (Hopper 1986, Martinez et al. 1999). In 1998, 4D Graphics for Construction Planning and Site Utilization

(4D-GCPSU) was developed by Zhang et al. (Zhang et al. 2000). The motivating interest behind 4D-GCPSU was to provide assistance at the site management level for planning and communication, a level which often requires flexible short term adjustment to plans in some details. Because of its Two-Way data exchange feature, users could access schedule management functions in the AutoCAD graphic environment, without switching to other schedule applications (Wang et al. 2004).

A newer information system platform, 4D Management for Construction Planning and Resource Utilization (4D-MCPRU), has been developed in order to implement the model 4DSMM+, which integrates dynamic resource management at the project level and decision making support with the other features (Wang et al. 2004). Besides 4D visualization, 4D-MCPRU tries to go beyond the first generation 4D tools, by extending the 4D model to the two pertinent fields of construction resource management and dynamic site planning.

4D WorkPlanner Space Generator (4D Space-Gen) was developed to automate the generation of work space requirements of activities. It was developed based on user-defined generic space requirement knowledge and project-specific production model information (Akinci et al. 2002).

Maa et al. (2005) developed a prototype called 4D Integrated Site Planning System (4D-ISPS) which integrates schedules, 3D models, resources and site spaces together with 4D CAD technology to provide 4D graphical visualization capability for construction site planning. In 4DISPS each component has a code, each task in WBS has a code too, and they can match with each other. In 4D-ISPS, standard objects are defined and drawn only once. If one standard object is constructed, all objects with the same properties can be generated automatically. In one word, WBS codes act as a bridge to link 3D models, schedules and other project information together (Maa et al. 2005). It should be noted that 4D ISPS does not have the function of position tracking of plants and common data standards that serve as the foundation for information sharing (Maa et al. 2005). Besides, the schedule it generates is a very basic one and needs modification by the user.

Mallasi (2006) established a new concept for “visualizing workspace competition” between the progressing activities through a number of objectives. PECASO was the name of the visual 4D CAD tool he developed. This dynamic 4D simulation environment was developed by utilizing three workspace planning features. Those are: (1)the twelve execution patterns, (2)the three different work rate distributions, and (3)a time-based simulation of the progressing quantities of work (Mallasi 2006).

The CVP system developed by Huang et al. (2007) allows project teams to check constructability, safety and to visualize 3D models of a facility before the commencement of construction works. The real life case study presented in the study shows that the CVP system is effective in assessing the executability of a construction planning including site layout, temporary work design, as well as resource planning (Huang et al. 2007).

Another effort was focusing on developing a new type of CAD system that automatically constructs and manages well-structured floor plans with minimum geometrical input from the designer. It also took advantage of the building data models developed from its prior researches. The model includes hierarchical building components such as ‘building’, ‘plan’, ‘space’, ‘ring’, ‘wall skeleton’, ‘surface’, ‘column’, etc. The creation algorithm developed assures a semantically rich and structurally correct floor plan at any point in the design process. In particular, the floor plan constructed through the design process contains spatial information as well as other design information about the building components. Thus, the system effectively manages spatial design information in the real-time basis (Jin et al. 2007).

In a collaborative effort by California Academy of Science, 3D, cost, and schedule of a building were put together to integrate the five dimensions of a museum project together. They used Graphisoft’s Constructor and Estimator software for space and cost modeling, and P3 for scheduling. A model manager at the top was coordinating these three modeler teams. Actually this was not an integrated 5D modeling effort since three different software packages were used by three different modeler teams. But it was an appreciable effort since it used a parallel process for analyzing the proposed plans.

Lee et al. (2003) define nD model as: an extension of the building information model, which incorporates multi-aspects of design information required at each stage of the lifecycle of a building facility. An effort was made in university of Salford, UK where they tried to come up with an nD modeling prototype. The 3D to nD research project at the University of Salford is developing a holistic nD modeling tool using IFCs, to help improve the decision making process and construction performance by enabling true 'what-if' analysis to be performed to demonstrate the real cost in terms of the variables of the design issues.

Critique of Conventional Planner Software and Prototypes

Study of mentioned software and prototypes showed that all of them get the model from the planner and provide illustration facilities such as 3D visualization, illustration of sequence of construction, site layout visualization, resource utilization, etc. to "show" how the project in reality is constructed.

Table 1

Software Category

Prototype / Software	**3D**	**Schedule**	**Cost**	**Resource**	**Constructability**	**Auto Decision Making**	**Error Detecting**	Commercial Package Example
3D modelers								Revit
4D Simulators								Synchro
5D Simulators								VICO
VP systems								CVP

Table 1 categorizes the studied prototypes and commercial packages which are being used in construction planning. It shows that none of the studied platforms in the literature automatically detects any error or makes a decision to propose it. Although in some prototypes, some types of resource allocation and schedule optimization are performed, what we are discussing here is beyond optimization of an already developed schedule or allocation of manually defined resources.

The second critique on most of either the prototypes and commercial packages or processes for integrating the various aspects of construction is that they are fragmented. At the best cases, an organizer at the top manually integrates the separated islands of scattered aspects of the project and bridges them together. In this pattern, when the planner / architect / engineer makes any change in one part, all of the parts would not be synchronized automatically. Human error is an Achill's heel to make the project error prone in this bottleneck.

Putting all different aspects (dimensions) of the model in one "land" is a better solution to this issue. Object-Modeling, which is the base of the proposed system, is the current application which could be used for this issue.

Before expressing the third critique, let's explain two phrases which might help more in understanding the issue. Here, "Optimum Plan" refers to a plan which satisfies the general requirements of the project in terms of time, cost, and constructability. "Desired Optimum Plan" is a plan that in addition to above mentioned general characteristics, it best fits specific needs of that specific contractor in that specific jobsite, with specific conditions and limitations.

Waly & Thabet (2002) argued and believe that most of the planning approaches impose a heavy burden on project teams due to the large amount of information and the interdependencies between different elements. All that discussed in the state of the art were based on this concept that "the planners must have a schedule in mind out of their past experiences on similar projects". Then they bring their idea into reality (actually into virtual reality), and nourishes their idea and analyze it in the above mentioned software packages to result in an optimized plan. Figure 1A illustrates this concept and shows the manual effort the planners must make to take the Optimum Plan to Desired Optimum Plan. But a critique applies to the rationale of this thesis: the starting plan is not necessarily a plan close to the optimum one. It is just based on the similar experiments which were not necessarily the optimum ones. This results in "having a long way ahead" to reach to the "desired optimum plan".

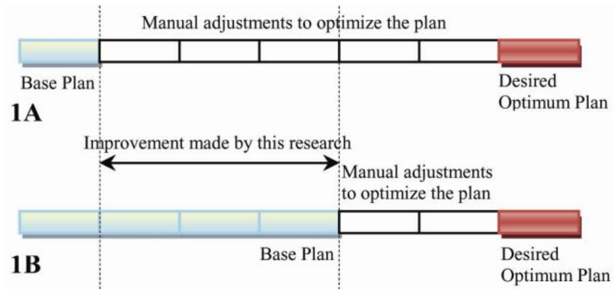

Figure 1: Progress in scheduling

In Figure 1B the goal of the current research is shown. The proposed system in this paper aims to automate a portion of this manual work by auto generating an optimum plan. The planner can set it as the basic plan and fine tune it manually based on the unique specifications of that project, the company, that period of time, etc. to reach to the Desired Optimum Plan sooner and easier.

The reason that the authors do not think of auto-generating a final Desired Optimum Plan is that simulation strategies are not expected to fit all construction projects, because of the uniqueness of construction operations (Hopper 1986, Martinez et al. 1999).

Therefore, the missing link in this chain is "where is the optimum point" to base the work on it. By setting the base on a point close to the desired optimum point, there would not be a long way ahead.

When there is a heavy burden of work on the planning phase, it is not unexpected that this "way" is not passed completely. The proposed method, by performing an integrated 5D simulation, aims to offer an "Optimum Plan" which is close to the "Desired Optimum Plan". From that point, the planner can perform minimal modifications (if needed) to take the "Optimum Plan" to the "Desired Optimum Plan".

The Proposed System

Despite most of the commercial packages which are just analysis tools instead of modeling tools the proposed system is considered to be a modeler as well. The other advantage is that it does not need another software package for adding the 4th dimension (time) and 5th dimension (cost) to the model. The proposed software is a library based software in which all the elements are pre-modeled and there exist in the library. The designer recalls them from the library and adds them to the model.

The objects in the model are parametric. One implication of the parametric is internal. It means some characteristics of an object are related with others, e.g., number of mullions is a parameter of the length of the window. The other implication of parametric is external. It means the object is defined in relation to the other objects. In the other words, an object is not an independent object in the model. It is dependent to the other objects and is defined as parameters of them. For example, an electrical outlet is defined as an object that is installed on gypsum board; and the gypsum board is a board which is screwed on stud and is painted; and the stud is a profile which is screwed to the runner and supports the gypsum board; and so on. However, these were not all the parameters defined in the mentioned objects. The external features of the parametric modeling help embedding the precedency of the objects in them. In this way, an object in the model has the identity of the object in the reality e.g. wall is wall because it has characteristics of a wall, not because it seats on the place of a wall in the model (basics of the parametric BIM software packages).

Each object is defined in more than one mode. One implication of this is about adjacency, e.g. a wall is defined to attach to either another wall or a column or just to a ceiling and floor without lateral supports. The other implication is about execution methods e.g. when calling a foundation from the library and putting it in place, it comes with a default formwork (e.g. wood) being shown in an interactive window. The modeler can leave it as default or can select another type of formwork in that interactive window (e.g. masonry).

Therefore, many parameters are defined in an object when it is in the library. When the object is re-called from the library to the model, some of the parameters (those related to adjacency) “catch” the related parameters in the adjacent objects, and the rest of the adjacency parameters, which are not applicable in that condition, would be inactivated temporarily. The same thing happens to execution parameters. In the previous example, when a wall is called from the library and put beside anther wall, those characteristics (defined as parameters in the objects) which represent the wall when it is beside a column or when it is just supported by floor and ceiling, all of them become inactive. For executing the foundation example, when wood formwork is selected as formwork, the parameters defining wood formwork become active and those defining masonry or dug-earth become inactive. Therefore, by making any modification in the model, e.g. putting a column at one side of the mentioned wall, some of the inactive parameters become active and vice versa.

The other characteristic embedded in the objects (through defining parameters) is precedency (sequence of constructing the objects). To make it a flexible plan, just mandatory predecessors are being forced and successors are not defined in order to be able to generate all feasible (constructible) plans. As an example, considering continuous columns up to 5th floor, installing beams of 4th floor is defined as predecessor of both pouring concrete of 4th floor and installing beams of 5th floor. But it does not force which one of these two are the successors of installing beams of 4th floor. In other word, successors are not defined in order to be selected flexibly, while predecessors are defined because they are mandatory for constructability. As a result, once beams of 4th are already set up, concrete of 4th floor could be poured before installing beams of 5th floor and vice versa.

Duration of performing the activity is the other characteristic defined in the objects. This is not a fixed duration. It is defined how long it takes to construct that object in different conditions. These different conditions come from the order of constructing the project (sequence of schedule), method of constructing the object, and production rate and number of crews. The number of crews, which is a coefficient of standard crews, is defined as a “default”, and the upper and lower limits are adjustable by the user in an interactive window upon recalling the object from the library when modifying it. It seems necessary to say that sequence of constructing the objects affects duration of constructing each of them, since access points, temporary equipments, and many other factors determine duration of a series of tasks. It is said that different methods of constructing the objects are embedded as parameters in the objects and such methods affect duration of constructing those objects. If the methods of constructing an object are known, duration of constructing the same object with the same equipment but different number of crews usually follow some simple rules of thumb. This idea could be implemented in defining the objects in the library when the programmer is defining the objects in the library. Otherwise, all of the combination of crew and equipments should be defined. However, based on the RSMeans, the number of such objects is not noticeable.

To be able to generate and evaluate all feasible ways of constructing an object, possible permutations of sequencing the tasks for the same object would be predefined in the objects in the library. When running the simulation, all of them, which are feasible, would be evaluated.

“Access canal” and “working space” are the other characteristics defined in the objects. For constructing each object, an approximate area is occupied around the objects for material, equipment, and crews. The temporary working space is defined with two parameters: “volume” and “penetration coefficient”. “Volume” is the volume of minimum geometrical space required for working space around the object. “Penetration Coefficient” is a factor which says how much of this volume is needed for the specified object, and how much of it could be shared for constructing the adjacent objects. Idea of defining a space around the objects is first developed in CIFE in 1998 (McKinney 1998). “Access Canal” which is defined as a temporary space for transporting material and equipment to their proper place during a specific period of time is useful since it considers constructability. Its importance is mostly shown up in big equipments (like AHU), and in the time they need to be placed at their proper location before closing the space around their destination location.

The other characteristic embedded in the objects is the cost of constructing that object. Like duration, this cost is not “one” fixed cost, but is driven by the method of constructing that object, number of crew, and equipment already defined and used for duration. Different combinations of crew and equipments would be defined and embedded as data in the objects.

Therefore, we have some objects which have different “ways” of construction and their associated installation predecessors and cost and time embedded in them through defining parameters. We can start construction of the building from different spots. The number of the spots is driven by the number of crews we afford/select as well as what planner decides. Max/Min of the number of crews are defined either by the user or the default of the program is being used.

Having all above mentioned data, the software will be able to automatically generate alternative plans for constructing the same project, and analyze them in five dimensions. This method can be explained by studying the method of storing data in the objects. It was discussed before, that each series of the parameters are defined for one execution situation. Those parameters can be put together in a “data bundle” embedded in the object. Each element of a data bundle is a parameter of the object, and each data bundle represents the object for one specific execution condition. For example if object “A” is represented by defining a, b, c, d, e, and f as different parameters, they could be defined in a data bundle.

This data bundle could be defined for different execution conditions of the same object (e.g. object “A”) based on its predecessors, adjacent objects, crews, equipment, and some other factors. Therefore, object “A” embeds “n” data bundles for “n” different conditions. This is the mode of object A when it is still in the library. When the object is recalled from the library and put into the model, some of these data bundles become “impossible or meaningless” and become inactive in the model (refer to the example of wall when it seats beside another wall). Therefore, when the model is already developed, the logic of the software runs the simulation and automatically generates alternative plans by passing the thread of plan through the objects in all different feasible orders (sequences). In each run, one data bundle of each object is taken into account and the rest of the data bundles are ignored. Selection of that data bundle is based on the affecting parameters like precedency, execution way, etc.

Run 1: 1, 3, 2, 6, 4, 7, 11, 12, 9, 8, 5, 10
Run 2: 1, 2, 6, 12, 7, 3, 11, 4, 8, 5, 10, 9
Run 3: 11, 12, 3, 2, 6, 1, 7, 10, 9, 8, 5, 4
Run …: ……..

Figure 2: Different plans of construction a three objects project

Figure 2 shows a project with three objects and 12 tasks to construct them. Although all 12! permutations of the tasks for generating 12! runs do not happen (because they do not lead to feasible plans), active data embedded in the objects allow some of them and a less number of runs would take place. Here it is assumed that no task can get done simultaneously with another one. Figure 3 shows one of the possible runs in the model.

Run 1: Tasks:	1	3	2	6	4	7	11	12	9	8	5	10
Run 1: Objects:	A	A	A	B	B	B	C	C	C	B	B	C

Figure 3: Example of repeating an object when making permutations of the tasks

At the end of each run, where the schedule thread is passed through one permutation of feasible order of constructing all the objects in the model, “m” different data bundles (m = number of objects in that model) would be generated, each one defining a series of parameters of one object of that thread (run). Cost, duration, and any other characteristics that planner needs would be extracted from the simulation. It should be noted that selecting a data bundle of an object does not mean “constructing the associated object completely at one time”. Selection of a data bundle means that the object is represented by that data bundle. That run of the simulation would deal with the characters of the object represented by that data bundle e.g. sequence of performing the tasks of that object would follow what the selected data bundle says, but some other tasks (belonging to the other objects) might be performed between them.

When all the feasible sequences of constructing the project are defined, the planner already has the characteristics of each thread defined. These characteristics are defined in L number of larger data bundles, in which L is the number of feasible planning threads (runs), and each larger bundle contains all the data of the data bundles of that run . It is possible that planner defines which characteristics of the project are needed to be extracted from the larger data bundle e.g. he can ask for duration, cost, cashflow, task order, etc. Based on the characteristics the planner needs as optimization criteria, an output bundle would be generated automatically in which the number of the elements is the number of the criteria that planner defined. Breakdown of each of them could be extracted from the corresponding element of this output bundle.

A function called “Utility Function” would be defined which entails the major factors of the planner such as time, cost, cashflow, etc. In this function, minimum cost, minimum duration, minimum deviation from cashflow, etc. could be the desired factors. The interaction of the factors is project-based and should be defined in each project separately. All generated output bundles in previous step would be taken into this function to find the output bundle which best satisfies the factors of this function all together. This function prioritizes the output bundles (actually plans) considering the criteria defined in the function. Figure 4 illustrates this part.

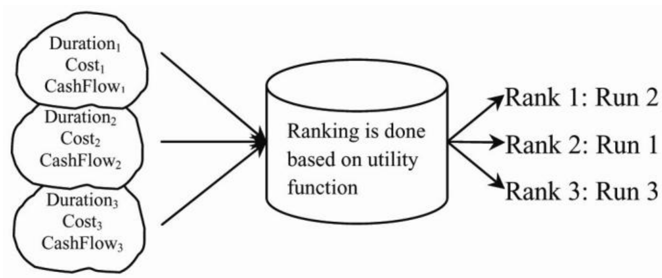

Figure 4: Utility function prioritize the plans

The planner can pick the top, top three, or top five threads, which are the proposed plans of the software, and set each of them as his “Optimum Plan” and review them in the current software packages. After all, he can manipulate one of them, or in some parts combine 2-3 of them to get to his “Desired Optimum Plan”, which best fits his company’s needs.

The other advantage is architect and those who do not have a thorough expertise in planning and estimating e.g. owner could easily have a general sense of project’s scheduling, duration and cost in each phase of the design. This system could easily reflect the effect of any change in design without going through manual process for attaching schedule and cost to each object. It would help the owner and architect in what-if-scenario analysis to find the best design.

Logic of the Proposed Software

predecessors are embedded in the objects, the software can list the “firsts”, those do not have any predecessor, in the objects’ list. Number of our crew and equipment determines how many tasks should be selected to get done in that step. It was explained that tasks are embedded in the object as parameters. Selection between the tasks in each step gets done randomly, and the balance will go to the “firsts” of the next step. This cycle runs until no task is left in simulation engine and the project finishes. Now one run is done and one plan is generated. This simulation runs as many times as the user defines, and the results go to a processor which extracts desired data from each thread (generated plan). These data are actually the criteria of utility function to determine the optimum plan.

Future Steps

Authors of the paper believe that unique nature of the projects does not allow the computer to make the final decision for planning. Any plan generated by the artificial intelligence to make the final decision needs fine tune up by the human intelligence to best fit to special needs of the project & company. But here, “how optimum” is matter. Referring back to Figure 1, the blank area, where needs manual modification by the user, could be minimized as much as possible. Proposed approach covers some parts and ignores some. Future steps are to cover the gaps of the proposed system.

This system ignores most of site access restrictions of the project. Considering a symmetric building, it does not make any difference for that system to approach it from east to west or from west or east. This system considers the overhead costs as a portion of direct costs of constructing the objects. Therefore, extra charges of interruptions during the project are ignored. It is expected that future researches cover these gaps and further develop the “Optimum Plan” and limit the manual work of planning (the blank part in Figure 1).

Conclusion

Literature shows that developed commercial software packages and academic prototypes provide the project planner with an environment in which they can analyze their plan in a virtual environment. Progress in those prototypes was directed towards providing the planner with more facilities in that environment to better and deeper analyze his proposed plan.

But the proposed plan which works as a base plan for improving to take it into a final plan, is rooted from the previous plans which were not necessarily an optimum one, and consequently the current plan would be used for future plans, while may not reach to its optimum point. The main rationale behind it is the huge work needed for reaching a regular plan to an optimum one, when a lot of factors affect the optimization. The supporting idea of the authors for claiming “non-optimum” for most of the projects is the huge number of change orders during construction phase.

The proposed method integrates five dimensions of the project with constructability issues in order to automatically generate almost all feasible methods of planning a construction project. Then it extracts the general information of each generated schedule. A utility function is the final part of decision support system which ranks the generated plans based on the defined criteria. The planner can now pick top two, three, or five plans as “optimum plans” and revise them in the current developed prototypes or commercial packages and by minimal modifications nourish them to take them to a “desired optimum plan”.

The authors believe this way shortens the way ahead of the planner for reaching to the “desired optimum plan” and will decrease the burden of work on planner’s shoulder, which results in a more accurate output.

References

Akinci, B., Fischer M., & Kunz, J. (2002). Automated generation of work spaces required by construction activities. *Journal of Construction Engineering and Management, 128* (4), 306–315.

Jin, C., Kwon, W., & Young, D. Hwang, Eun, J., & Lertlakkhanakul, J. (2007). Real-time management of spatial information of design: A space-based floor plan representation of buildings. *Automation in Construction, 16,* 449–459.

Cleveland, Jr. A.B. (1998). Real-time animation of construction activities. *Proceedings of Construction Congress I—Excellence in the Constructed Project,* 238– 243.

Collier, E., & Fischer, M. (1995). Four-dimensional modeling in design and construction. *CIFE Technical Report, No. 10.* Stanford University, Stanford, CA.

Fischer, M., Hartmann, T., Rank, E., Neuberg, F., Schreyer, M., Liston, K., & Kunz, J. (2004). Combining different project modeling approaches for effective support of multi-disciplinary engineering tasks, in: Brandon, P. Shaffii, H. Li, Shen (Eds.), Q. *INCITE 2004 — International Conference on Information Technology in Design and Construction, Langkawi, Malaysia, 167*–182.

Heesom, D., & Mahdjoubi, L. (2004). Trends of 4D CAD applications for construction planning. *Construction Management Economics, 22,* 171–182.

Hopper, J. W. (1986). Strategy related characteristics of discrete-event languages and models. *Simulation, 46*(4), 153–159.

Huang, T., Guo, K., Baldwin, A., & Li, H. (2007). A virtual prototyping system for simulating construction processes. *Automation in Construction, 16,* 576–585.

Huber, B., & Reiser, P., (2003). The marriage of CPM and lean construction, in: J.C. Martinez, C.T. Formoso (Eds.). *11th International Group for Lean Construction Conference*. Virginia Tech, Virginia, USA.

Jongeling, & R., Olofsson, T. (2007). A method for planning of work-flow by combined use of location-based scheduling and 4D CAD. *Automation in Construction, 16,* 189–198.

Kahan, E.T., & Madrid, X.H. (1987). Integrated system to support plant operations. *Hydrocarbon Processing Symposium,* 55– 60.

Lee, A., Marshall-Ponting, A.J., Aouad, G., Wu, S., Koh, I., Fu, C., Cooper, R., Betts, M., Kagioglou, M., & Fischer, M. (2003). *Developing a vision of nD-enabled construction*. Construct IT Report, ISBN: 1-900491-92-3.

Maa, Z., Shena, Q., & Zhang, J., (2005). Application of 4D for dynamic site layout and management of construction projects, *Automation in Construction, 14,* 369–381.

Mallasi, Z. (2006). Dynamic quantification and analysis of the construction workspace congestion utilizing 4D visualization. *Automation in Construction, 15*, 640 – 655.

Mallasi, Z., & Dawood, N. (2002). Registering space requirements of construction operations using site-PECASO model, in: Agger, K. Christiansson, P. Howard (Eds.), R. *CIB w78 conference 2002 — Distributing Knowledge in Building,* Aarhus School of Architecture, Denmark, 1–8.

Martinez, J., & Ioannou, P. G., (1999). General-purpose systems for effective construction simulation. *Journal of Construction Engineering and Management, 125*(4), 265–276.

McKinney, K., Kim, J., Fischer, M., & Howard, C. (1996). Interactive 4DCAD. *Proceedings of the Third Congress in Computing in Civil Engineering, ASCE, Anaheim, CA,* 383–389.

McKinney, K., & Fischer, M. (1998). Generating, evaluating and visualizing construction schedules with CAD tools. *Automation in Construction, 7*, 433–447.

Rischmoller, L., Fischer, M., Fox, R., & Alarco´n, L. (2001). 4D planning and scheduling: grounding IT research in industry practice. *Proceedings of CIB w78 2001 Conference on Construction Information Technology, Mpumalanga, South Africa*, 34/1– 34/11.

Waly, A.F., & Thabet, W.Y. (2002). A virtual construction environment for preconstruction planning. *Automation in Construction 12*, 139–154.

Wang, H.J., Zhang, J.P., Chau, K.W.,& Anson, M. (2004). 4D dynamic management for construction planning and resource utilization. *Automation in Construction 13,* 575– 589.

Williams, M. (1996). Graphical simulation for project planning: 4Dplanner. *Computing in Civil Engineering,* 404–409.

Zhang, J.P., Anson, M., & Wang, Q. (2000). A new 4D management approach to construction planning and site space utilization. *Computing in Civil Engineering,* 15–22.