14th International Conference on Computing
in Civil and Building Engineering
Moscow, Russia, 27 - 29 June 2012

Advanced spatio-temporal validation of construction schedules

Abstract

Visual 4D modelling and planning technologies are becoming increasingly important in complex construction programmes facing the problems of advanced communication among stakeholders, better utilization of critical resources, and effective spatio-temporal coordination of works. Popular 4D tools and systems provide basic functionalities to simulate project schedules in virtual environments and to identify simple conflicting situations caused by collisions and interferences of construction elements and equipment units. Due to their complexity the collisions are usually detected in pseudo-dynamic mode assuming all the changes occurring in discrete time moments. Ultimately, it enables to anticipate and avoid potential problems at earlier phases and to reduce risks and waste at final construction phases often undergone to delays and reworks. The aim of this paper is to systemize possible spatio-temporal conflicts and to present advanced methods for more comprehensive and trustworthy validation of project schedules. For this purpose, an extended test suite is proposed by composing four complementary groups, namely: clash, join, workspace and path tests. Compared to usual clash testing, the introduced test suite helps to identify non-trivial defects like missing of supporting neighbouring elements, unavailability of required workspaces and absence of collision-free paths to deliver the elements to destination locations. For each group of tests the formal mathematical criteria and efficient computational strategies are presented and discussed. It's essential that they do not need detailed specifications of testing use cases and can be applied for large-scale construction projects simulated in pseudo-dynamic mode. Conducted computational experiments have proved the effectiveness and the feasibility of the proposed 4D planning and validation methods.

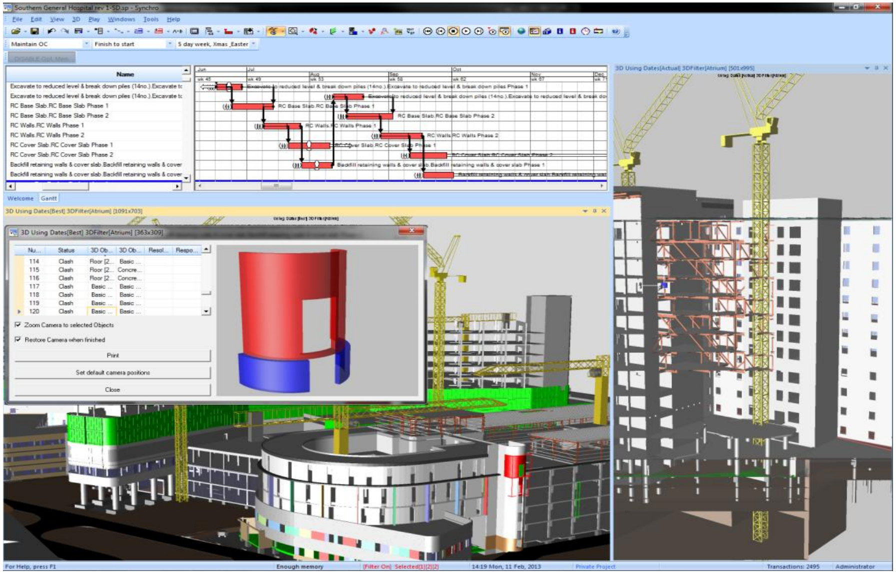
Keywords: 4D modelling, project planning and scheduling, validation, collision detection, path planning.

1 Introduction

Visual 4D modelling and planning technologies are becoming increasingly important in complex construction programmes facing the problems of advanced communication among stakeholders, better utilization of critical resources and effective coordination of works taking into account both: spatial and temporal aspects. 4D tools provide a more comprehensive multidisciplinary analysis of the planned project activities by consolidating both 3D CAD models and scheduling information delivered from the project management systems like MS Project, Primavera Project Management, Asta Powerproject.

As a result, these tools have a tremendous potential to increase the communication efficiency and interpretation ability of the project team members (Dawood and Sikka, 2008). Improved communications are reached as a result of the simulation of project activities ‘in progress’ and the visualization of the construction programme as an animated scene reproduced with using graphic facilities or virtual reality environments. Another major benefit is that the 4D tools allow planners to trade off the temporal sequencing of tasks with their spatial distribution, resulting in a more robust and rehearsed project schedule (Tulke and Hanff, 2007). With the increasing pressure for shorter delivery schedules, a better utilisation of space resource on construction sites becomes more apparent. As opposed to the traditional Critical Path Method (CPM) widely employed by popular planning tools, the Critical Space Analysis (CSA) emphases the dynamic spatial distribution of the activity execution. This concept has been successfully adopted by the 4D modelling tools based on the industrial requirements capture (North and Winch, 2002).

Popular 4D modelling systems like Synchro, Autodesk Navisworks, Bentley Schedule Simulator, Intergraph Schedule Review provide basic functionalities for simulating project activities in space dimensions and across time. Because of the complexity, projects are usually simulated in the pseudo-dynamic mode under common suggestion that most, if not all, objects appear, disappear or move strongly in the discrete time moments in which the project activities usually start or finish. Continuous behaviour for some animated objects is allowed, but the dynamic analysis of the whole scenes simulating large-scale industrial projects at detailed aggregation levels looks unrealistic. It is explained by intensive computations needed to carry out such analysis, as well as by enormous efforts to specify all the trajectories and kinematic rules, the construction elements and equipment units can move accordingly.



Maintain OC
Finish to start
5 day week, Xmas ,Easter

Name
Excavate to reduced level & break down piles (14no.).Excavate to
Excavate to reduced level & break down piles (14no.).Excavate to
RC Base Slab.RC Base Slab Phase 1
RC Base Slab.RC Base Slab Phase 2
RC Walls.RC Walls Phase 1
RC Walls.RC Walls Phase 2
RC Cover Slab.RC Cover Slab Phase 1
RC Cover Slab.RC Cover Slab Phase 2
Backfill retaining walls & cover slab.Backfill retaining walls & cover
Backfill retaining walls & cover slab.Backfill retaining walls & cover

Nu...	Status	3D Ob...	3D Ob...	Resol...	Respo...
114	Clash	Floor [2...	Basic ...		
115	Clash	Floor [2...	Concre...		
116	Clash	Floor [2...	Concre...		
117	Clash	Basic ...	Basic ...		
118	Clash	Basic ...	Basic ...		
119	Clash	Basic ...	Basic ...		
120	Clash	Basic ...	Basic ...		

☑ Zoom Camera to selected Objects

☑ Restore Camera when finished

Figure 1: The Synchro graphic user interface for visual representation of the 4D project data and spatial coordination.

Figure 1 illustrates the Synchro 4D modelling application: the traditional Gantt chart is combined and coordinated with multiple 3D views, simple clashes are enumerated in the clash test report.

The available 4D modelling systems are also capable of identifying simple clashes caused by collisions and interferences of construction elements to be installed in the same place at the same time. Nevertheless, such analysis, being applied to the pseudo-dynamic mode, omits many other important issues leading to potential conflicts at project sites. Therefore, the construction projects are needed in more comprehensive and trustworthy methods of validating design accuracy and schedule adequacy. Ultimately, the methods would enable to anticipate and to avoid potential problems at earlier project phases and reducing risks and waste at the final construction phase often being undergone to delays and reworks.

The objective of this paper is to present the advanced validation methods that would identify both: usual clashes and more sophisticated spatio-temporal defects of project schedules. For this purpose, an extended test suite is proposed by composing four complementary groups, namely: clash, join, workspace and path tests. Compared to the usual clashes, the test suite enables to identify non-trivial conflicting situations, like missing of supporting neighbouring elements, unavailability of required workspaces and absence of collision-free paths to deliver the elements to destination locations. For each introduced group of tests the formal mathematical criteria and efficient computational strategies are presented and discussed. It's essential that they do not need detailed specifications of testing use cases and can be applied for large-scale construction projects simulated in the pseudo-dynamic mode. Being applied concordantly, the proposed 4D validation methods help to identify and to resolve suspicious issues of the prepared project schedules and to rise up the trustworthiness of whole construction programmes.

The rest of the paper is organized as follows: in Section 2 we describe the peculiarities of the visual scenes appearing in 4D modelling and planning applications and introduce four groups of tests to validate project design and schedules against the potential spatio-temporal defects. Special attention is paid to the mathematical criteria and efficient computational methods for performing such tests for the large-scale construction project data. Meaningful examples are presented to explain each of the introduced group of tests. Their benefits are shortly summarized in Conclusions.

2 Validation tests

The considered test suite is feasible for the scenes originating from 4D modelling and planning applications and having the following characteristics:

Large scale: the scenes may consist of thousands and millions of objects with their own 3D model representations and dynamic behaviours. The objects can be both: relatively simple shapes and assemblies with sub-assemblies, as a result of which the complexity of individual objects and scenes can be essentially varied.

Mixed geometry: the objects may be canonical geometry primitives, algebraic implicit and parametric surfaces, like quadrics, NURBS and Bezier patches, convex and non-convex polyhedrons, solid bodies given by constructive solid geometry (CSG) or boundary representation (BREP).

Pseudo-dynamics: All the scene events are discrete in time and known in advance (in contrast to the real-time simulation in the virtual reality environments). They may be appearance or disappearance of the scene objects, as well as their discrete movements. The dynamic simulation of the whole scenes looks redundant and unrealistic for real construction projects. Nevertheless, it is admitted that some part of the objects can move smoothly along the specified trajectories in accordance with the prescribed kinematic rules.

The test suite is composed of four main groups that can be checked using the so-called clash, join, workspace, and path tests. Clash tests enable to identify simple defects by checking for contact between a pair of scene objects. Join tests complement these checks by exploiting the following evident principle: the construction elements cannot be correctly installed when isolated from the supporting or neighbouring elements. Workspace tests focus on the feasibility of scheduled activities which can be successfully carried out only if other concurrent activities are not running in the same space. Path tests are intended to guarantee the possibilities to deliver each construction element to the assigned destination position along a collision-free path avoiding any obstacles and satisfying the imposed kinematic constraints.

Clash, join and workspace tests can be performed using well-known mathematical methods of collision detection. Although these methods belong to the traditional chapters of the computational geometry and are incorporated in the most popular CAD and computer graphics systems, the performance remains a crucial factor for analysis of complex dynamic scenes, particularly, the scenes originating from 4D modelling and planning applications. Path tests can be accomplished using the theory of motion planning and its numerous applications. Unfortunately, both: collision detection and motion planning methods have relatively high complexity that grows extremely with the input data volume. Therefore, efficient computational strategies must be developed and applied to perform the proposed tests on large-scale construction project data. Let's discuss each of the introduced group of tests in more details.

2.1 Clash tests

Collision and interference checks constitute the first group. The problem of collision detection or contact determination between two or more objects is fundamental for computer animation, physical based modelling, CAD/CAM applications, robotics and automation, computer graphics, and virtual reality as well. The survey of the traditional methods and available tools can be found in (Lin and Gottschalk, 1998).

It is said that two objects o' and o'' don't collide with each other if the distance between them is larger than the given nonnegative threshold: $dst(o', o'') > \varepsilon_0 \geq 0$. Here the function is defined as a minimum Euclidian distance among all the pairs of points $x' \in o'$ and $x'' \in o''$ belonging to the corresponding objects: $dst(o', o'') \equiv \min_{x' \in o',\, x'' \in o''} \|x', x''\|$. The threshold ε_0 plays the role of the absolute computational tolerance with which the collisions are determined. Figure 2a presents an example of the clash testing with a given threshold parameter. It identifies the issue for circle O_1 and rectangle O_2, but proves the avoidance of collisions between circle O_1 and rectangle O_3.

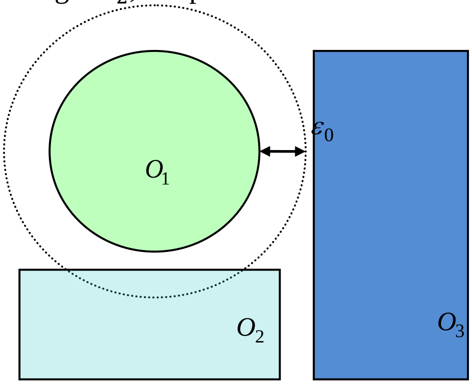


Figure 2a: An example of clash testing.

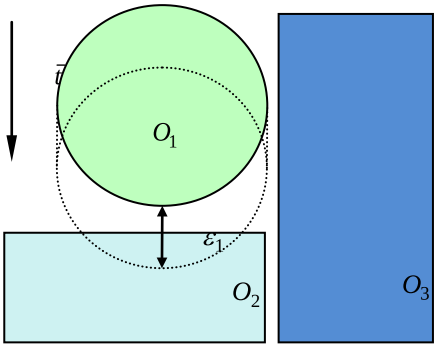


Figure 2b: An example of join testing.

Four fundamental approaches to the collision detection problem have been proposed and successfully implemented to deal with different statements and application peculiarities. These are exact interference detection (Lin and Gottschalk, 1998), spatial decomposition, bounding volumes techniques (Klosowski 1998; Zachmann, 1994) and methods exploiting temporal coherence (Jimenez et al., 2001; Cameron, 1990). They all focus on reducing the number of pairs of objects that need to be checked for contact, as well as on reducing the total computation cost of such checks. An efficient computational strategy for collision detection was proposed and investigated in our work (Semenov et al., 2010). Combining the mentioned above approaches it yields substantially faster collision detection than the previously known methods applied separately or discordantly.

2.2 Join tests

The second group of the validation tests is the so-called "join checks". They imply checking for contact between the new objects that appeared in the scene and the existing objects located in the same or nearby positions. Indeed, a newly installed construction element cannot "hang in the air" and must be supported by lower elements or fasten together neighbouring elements. If an object is removed from the scene, then the remained objects have to be suffered to join checks as they could be based on the removed object. To perform join tests, they must be preliminary specified in a form allowing mathematically strong validation. Let's discuss how such a specification can be compiled being based on the introduced adjacency relation among objects of a scene S.

General specification of joint tests should include the set of objects $S^* \subseteq S$ considered to be priori installed correctly. By pointing out such objects, we define initially deployed elements of the scene S, such as ground, stationary infrastructure, etc. The specification may also contain information about particular objects and additional requirements assuming the availability of neighbours for the installed objects in their final destination positions.

In most practical cases, a simple mathematical model can be utilised for this purpose. As suggested, an object $o' \in S$ is installed correctly and satisfies to corresponding join test if some object $o'' \in S$ has been already installed correctly at the distance not far from the given distance threshold: $dst(o', o'') \le \varepsilon_1$. In some cases, the directions should be additionally prescribed to particular objects to constrain the admitted location domains of neighbouring objects. To be applied concordantly, tolerance parameters of clash and join tests must be chosen in a proper way: $\varepsilon_0 < \varepsilon_1$.

An object $o' \in S$ is adjacent to an object $o'' \in S$ (or $o' \to o''$) in the specified direction $\bar{t}$ at the distance ε_1 if and only if exists vector $\overline{t'}$ collinear to $\bar{t}$ so that its length is smaller than the given distance ($\|\overline{t'}\| \le \varepsilon_1$) and the object o' being translated by the vector $\overline{t'}$ collides object o''. In practice to identify the adjacency between objects o' and o'', object o'' should be checked against the collision with object o''' obtained by extrusion of the object o' along vector $\varepsilon_1 \bar{t}/\|t\|$. Zero threshold parameter is assumed to be applied to this check. If the collision is identified, then object o' is adjacent to object o''. Figure 2b presents an example of join testing. Circle O_1 is identified to be adjacent to rectangle O_2 with the specified direction $\bar{t}$ and the given threshold ε_1, but not adjacent to rectangle O_3.

2.3 Workspace tests

To provide a safe and productive environment, project managers need to plan the work spaces required by construction activities. Work space planning allows different interpretations and covers meaningful statements like site layout planning, space scheduling, and space occupation balancing. (Akinci and Fischer, 1998; I-Chen Wu and Yen-Chang Chiu, 2010; Sy-Jye Guo, 2002).

Since construction schedules may consist of hundreds and thousands of activities requiring multiple types of spaces, it is practically impossible to expect project managers to specify manually all the data necessary for representing workspaces in complex scenes. At the same time, semi-automatic techniques are able to generate spaces using construction templates including their relative orientation with respect to a reference construction element and requiring a certain size (Akinci, Fischer2, Kunz 2000).

In this section we consider two underlying methods of defining and performing workspace tests. The first method corresponds to the space scheduling statement mentioned above. It implies defining an exclusive workspace, $s_i = S$, $i = 1..n$ for each schedule activity a_i and corresponding construction element $o_i \in S$ installed, removed or moved during this activity. A correct schedule must avoid conflicting situations when workspace of one activity is crossed by elements or workspaces of other running activities, thereby satisfying the conditions $dst(s_i, o_j) > \varepsilon_0$ and $dst(s_i, s_j) > \varepsilon_0$ for any $o_i, o_j \in S$, $s_i, s_j \in S$, $i \neq j$. This would mean that concurrent activities must not share the same spatial resource during the common time interval. Each workspace may be represented as a simple box or a set of solids reproducing different concepts and requirements, like labour crew space, equipment space, hazard space. It is admitted that objects and spaces belonging to the same activity can be mutually intersected, but situations of intersections with other spaces must be excluded.

Figure 3 illustrates an example of workspace testing. Objects $O_1$and O_2 have been assigned to the activities A_1 and A_2 assuming availability of free spaces S_1 and S_2 necessary for their successful performing. Since activities A_1 and A_2 share the common time interval (t_0, t_1) and spaces assigned to these activities intersect in the region S_{12} , spatio-temporal collision is being identified.

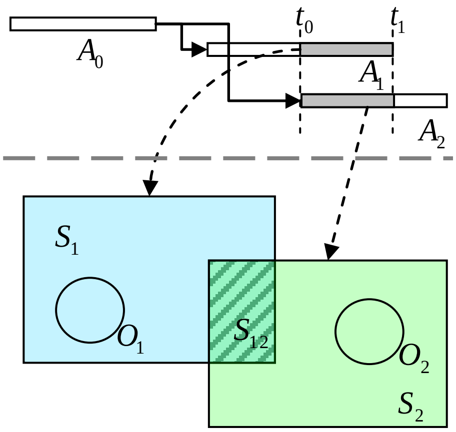


Figure 3: Space testing example.

The second method relates more to the space occupation balancing. The line of balance is a conceptual plot widely used in project management systems to visualize resource utilization degrees across time. The proposed method assumes the workspaces to be non-exclusive spatial resources shared by different activities simultaneously. Certainly, the specification of all admitted combinations of human activities, construction methods and technologies which could be performed in the given workspace looks extremely difficult to project managers. Therefore, we propose to use the following mathematical model for space sharing. It is formally expressed by pseudo-Boolean functions of the form $f: B^n \to R$, where $B = \{0,1\}$ is a Boolean domain, R is a real domain and n — is a nonnegative integer called the arity of the function.

Let's introduce a Boolean indicator $a(t) \in B^n$ indicating which activities started, but not finished at the given time moment t so that for every activity a_i the indicator element $a_i(t) = 0$ if the activity is not running and $a_i(t) = 1$ in the opposite case. Then, a generic constraint for workspace utilization can be represented by a multi-linear polynomial function as follows:

$$f(a,t) = \sum_i b_i a_i(t) \leq 1$$

where all the polynomial coefficients $0 \leq b_i \leq 1$ are normalized. It can be seen that the presented constraint enables to specify various meaningful cases of workspace utilization.

If a workspace is exclusive, then the coefficients can be defined as $b_i = 1$ for all the activities a_i, $i = 1..n$. The constraint is satisfied only if the activities are executed one after another and it is violated if some activities are running at the same time. If an activity a_i is allowed to run concurrently with other activities a_j, $i \neq j$, then corresponding coefficient must be set to zero value $b_i = 0$. If a workspace can be shared by all the activities simultaneously, then the constraint takes the confluent form with all the zero coefficients $b_i = 0$, $i = 1..n$. Certainly, the coefficients may not necessarily be integer. More complicated cases are covered by using real values. For example, if $b_i = 1/2$, $i = 1..n$, then any pair of activities is admitted to be run simultaneously; if $b_i = 1/3$ — any three activities, etc. Thus, the introduced coefficients define fractions of the common workspace utilized by each involved activity and can be adjusted individually.

Being specified and interrelated with involved activities, the workspaces can be tested against imposed constraints. The computational methods needed to perform such tests are similar to those applied for clash tests with the exception that both construction elements and related workspaces are suffered to collision analysis. Certainly, pairs of elements and workspaces belonging to the same activities should be excluded from the consideration.

2.4 Path tests

The fourth proposed group of validation checks is path tests. These tests are intended to control the possibilities to deliver each construction element or equipment unit to its destination positions, or in other words, the existence of collision-free paths from some initial outdoor position to the final installation position. For removed elements the existence of paths from installation positions to the outdoor position is checked too. Path tests make sense only for those scene objects whose continuous behaviour has not been specified exactly. Clash and interference checks performed in pseudo-dynamic mode can guarantee absence of collisions only in discrete time moments rather than over whole time intervals when the objects are moving. Therefore, clash testing is not comprehensive and complementary path tests would add value to the evolved validation technology. For the animated objects moving along specified trajectories in accordance with the prescribed kinematic rules, usual clash testing enables to identify all the discussed critical issues.

So, if an object $o_i \in S$ appears in the scene S at a fixed time moment, we require the existence of a collision-free path $p(\tau)$, $\tau \in [0; 1]$ from the predefined outdoor position $p(0) = x_0$ to a destination position $p(1) = x_1$ so that the object o_i, being placed in any intermediate position and represented as $o_i' = p(\tau) \circ o_i$, does not collide with other scene objects $dst\left(p(\tau) \circ o_i, o_j\right) > \varepsilon_0$, for any $i \neq j$ and $\tau \in [0; 1]$. Note that this definition admits the object o_i to be translated and rotated, but neglects the speed with which the object can move under accepted assumptions. Figure 4 presents an example of path testing for object O being moved from the position A to the destination position A'. Among possible routes like P_1 and P_2 , the collision-free path P_2 has been found. Evidently, route P_1 leads to the clash between the object and the environment.

The presented statement relates to the global path planning problem. Unlike local planning, it has relatively high computational complexity that extremely grows with the input data volume. Extensive research efforts have been directed towards these problems (LaValle 2006). Most reports have concluded that the algorithms work well in simple 2D environments, but require much larger computation resources in large-scale dynamic 3D environments. It makes the discussed validation tests highly intractable for the construction planning applications.

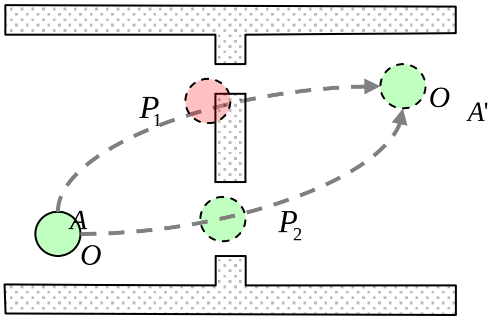


Figure 4. Path testing example.

Effective computational approach could consist in a combination of topological and metric schemes leveraging both global and local planning strategies (Lamarche 2009). Topological schemes are used for making high-level decisions about perspective routes, and metric schemes — for local correction of routes and their final validation. The approach would provide a whole coverage of complex indoor/outdoor environments and would resolve multiple requests in reasonable time. Some particular methods have been developed in the scope of this approach (Ellips and Davoud 2007). Conducted computational experiments proved their suitability to the discussed validation problems.

3 Conclusions

Thus, advanced methods for a more comprehensive and trustworthy validation of project schedules have been presented. They assume the extended test suite composed of four complementary groups of checks, namely: clash, join, workspace and path tests. Compared to the usual clash testing, the test suite helps to identify non-trivial defects like missing of supporting neighbouring elements, unavailability of required workspaces and absence of collision-free paths to deliver the elements to the destination locations. For each group of tests the formal mathematical criteria and efficient computational strategies are presented and discussed. It is essential that none of them need detailed specifications of testing use cases and can be applied for large-scale construction projects simulated in the pseudo-dynamic mode. The conducted computational experiments have confirmed the effectiveness and the feasibility of the proposed 4D planning and validation methods, which looks very promising when used in the industry practice.

4 REFERENCES

AKINCI B., FISCHER M., and ZABELLE T., 1998. Proactive Approach for Reducing Non-value Adding Activities Due to Time-space Conflicts. *Proc. 6th Ann. Conf. Lean Constr., Guaruja, Brazil, August*, 12 pp.

AKINCI B., FISCHER M., KUNZ J., 2000. Automated Generation of Work Spaces Required by Construction Activities. *Journal of Construction Engineering and Management, Volume 128, No. 4*, pp. 306-315

CAMERON S., 1990. Collision detection by four-dimensional intersection testing. *IEEE Trans. Robot. Autom., 6(3)*: pp. 291–302.

DAWOOD N. and SIKKA S., 2008. Measuring the effectiveness of 4d planning as a valuable communication tool, *ITcon, Vol. 13 (2008)*, pp. 620-636.

ELLIPS M., DAVOUD S., 2007. Classic and Heuristic Approaches in Robot Motion Planning - A Chronological Review. *Proceedings of world academy of science, engineering and technology 23*, pp. 101-106.

I-CHEN WU., YEN-CHANG CHIU, 2010. 4D Workspace Conflict Detection and Analysis System. *Proceedings of 10th international conference on construction applications of virtual reality.*

JIMENEZ P., THOMAS F., and TORRAS C., 2001. 3D collision detection: a survey. *Computers and Graphics, 25*, pp. 269–285.

KLOSOWSKY J.T., HELD M., MITCHELL J.S.B., SOWIZRAL H., ZIKAN K., 1998. Efficient collision detection using bounding volume hierarchies of k-DOPs. *Visualization and Computer Graphics, IEEE Transactions on Volume 4, Issue 1* pp, 21 – 36.

LAMARCHE F., 2009. TopoPlan: a topological path planner for real time human navigation under floor and ceiling constraints. *Computer Graphics Forum. 28 (2).* pp. 649-658.

LAVALLE S. M., 2006. *Planning algorithms.* Cambridge University Press, 2006, 842.

LIN M. and GOTTSCHALK S., 1998. Collision detection between geometric models: a survey. *In Proceedings of IMA, Conference of mathematics of surfaces*, 1998, pp. 602–608.

NORTH S. and WINCH G.M., 2002. VIRCON: A proposal for critical space analysis in construction planning, *Proceedings of the fourth European Conference on Product and Process Modelling in the Building and Related Industries (ECPPM), Portoz, Slovenia*, pp. 359-364.

SEMENOV V.A., KAZAKOV K.A., ZOLOTOV V.A., 2010. Combined strategy for efficient collision detection in 4D planning applications. *In Computing in Civil and Building Engineering, Proceedings of the International Conference, W. Tizani (Editor), UK,* Nottingham University Press, 2010, pp. 31-39.

SY-JYE GUO., 2002. Identification and Resolution of Work Space Conflicts in Building Construction. *Journal of construction engineering and management, 2002*, pp.287-295.

TULKE J. and HANFF J., 2007. 4D construction sequence planning – new process and data model. *CIB W78, 24th Conference, Maribor.*

ZACHMANN G., 1994. Exact and Fast Collision Detection. Diploma thesis, Fraunhofer Institute for Computer Graphics Technische Hochschule Darmstadt, Fachbereich Informatik, Germany.