MODELING AND SIMULATING SPATIAL REQUIREMENTS OF CONSTRUCTION ACTIVITIES

Arnim Marx
Markus König

Chair of Computing in Engineering
Faculty of Civil & Environmental Engineering
Ruhr-Universität Bochum
Universitätsstr. 150
Bochum, 44780, GERMANY

ABSTRACT

Spatial conflicts often occur on construction sites and can have significant impact on the total construction progress. However, the specification and calculation of spatial requirements for construction processes is very time-consuming. For this reason, spatial aspects are often not considered adequately in practice. In this paper an approach for modeling spatial requirements of construction activities using building information models is introduced. Based on the Partial Model Query Language (PMQL) some extensions have been developed to define spatial requirements efficiently. To identify and solve spatial conflicts, the spatial requirements can be integrated into a constraint-based simulation approach. During the simulation run, processes are evaluated regarding their spatial requirements. If spatial conflicts are detected, processes are rescheduled accordingly by increasing process durations or postponing start dates.

1 INTRODUCTION

Today, construction companies have less and less time for planning and construction due to the constantly rising competition in the construction industry and the ever increasing complexity of the construction projects. Thus, an accurate and detailed scheduling is required to perform as many construction activities simultaneously as possible and utilize the equipment and workers optimally. The parallelization of construction works as well as high employment of resources often leads to spatial conflicts, since the working space is limited on the construction site. A spatial conflict arises when the spatial requirements of construction tasks which are executed at the same time overlap. Particularly during finishing works spatial conflicts occur due to a high number of different teams working simultaneously. In Sanders (1989) it was stated that spatial conflicts disturb the execution of operations and reduce on-site productivity significantly. Riley and Sanvido (1997) observed a construction site for two months and recognized 71 spatial conflicts regarding four trades. To minimize such spatial conflicts, the spatial requirements of each construction activity need to be taken into account during scheduling. However, this a very challenging task due to the change of the building and construction site over the time as construction proceeds.

In the last years, discrete event simulation has been applied to analyze complex construction projects more and more often (AbouRizk 2010). Using a simulation model the project duration can be forecasted, resource allocation can be generated, transport ways and storage areas can be planned and the material flow can be optimized. Few applications are dealing with geometric or spatial aspects. However, usually only specific operations like crane movements or assembling works are examined exemplarily. As a consequence of the availability of three-dimensional building information models a multitude of spatial in

formation is available. However, there is a lack of suitable concepts to use building information models within simulation models to analyze and reduce spatial conflicts during scheduling.

In this paper an approach to model and consider spatial constraints of construction works by using building information models is presented. Concepts have been developed to extract, generate and simulate spatial requirements of construction activities.

2 RELATED WORK

Several concepts exist to model and consider spatial requirements in the context of construction scheduling. Most researchers focused on analyzing a given construction schedule regarding spatial conflicts. Tommelein et al. introduced a concept to model spatial requirements using elemental two-dimensional shapes like rectangles or circles (Tommelein and Zouein 1993; Choo and Tommelein 1999; Zouein and Tommelein 2001). Using a given schedule time-dependent spatial conflicts can be detected. Afterwards, manual adaptions are performed to solve the identified conflicts. Akinci et al. developed the 4D Work-Planner Time-Space Conflict Analyzer to simulate construction works under consideration of different spatial requirements (Akinci and Fischer 1998; Akinci et al. 2002). Spatial requirements can be modeled by Axis-Aligned Bounding Boxes (AABB) regarding reference objects of a so-called space-loaded production model. Discrete event simulation is applied to check the spatial requirements and identify spatial conflicts. However, solving spatial conflicts is still a manual task.

Current approaches apply collision detection tools to detect spatial conflicts (Kamat and Martinez 2005; Lai and Kang 2009). Chavada et al. (2012) used the XNA game engine based on AABB spaces to calculate collisions of a given construction schedule. Spatial requirements of the construction activities were modeled manually considering building information models based on the Industry Foundation Classes. Detected collisions can be visualized and stored in a database for further investigations and manual adaptions. Similar concepts are used by Mawlana et al. (2012) based on the simulation software STROBOSCOPE and the clash detection functionality of Autodesk Navisworks.

The generation of construction schedules without spatial conflicts were addressed by a few researchers. Akbas (2003) introduced a concept for simulating construction processes as sequences of crews acting on geometric work locations. That means that only one crew can work at a certain work location at a time. Furthermore, a production rate function was specified to calculate at what rate crews can perform work at a specific work location at a specific time. The production rate depends on other crews which are acting on nearby work locations. Work locations are modeled based on a two-dimensional tessellation of the construction site. Spatial conflicts can be solved by using different workflow strategies of the crews. Zhang et al. (2005) developed a cell-based simulation concept to generate schedules without spatial conflicts. To achieve this, the building site is divided into two- or three-dimensional regular cells. Conflict detection was done by checking the state of each cell and avoiding the use of a cell by more than one object (Zhang et al. 2005). However, using cells to model and check spatial requirements of large construction sites seems to be computationally expensive and less accurate.

3 MODELING SPATIAL CONSTRAINTS

In this paper a BIM-based approach to model and simulate spatial requirements of construction activities is proposed. According to Riley and Sanvido (1995) three different types of spatial requirements are distinguished: resource, site and activity requirements. However, most interesting are spaces required by activities and their involved resources. A typical example is the space which is required to install a window using a hydraulic lift (cf. Figure 1d). The required space consist of the lift space (cf. Figure 1a), the working space of the installation crew (cf. Figure 1b), and the safety area under and round the lift (cf. Figure 1c). It is obvious that the specification of complex and project-specific spatial requirements is very time consuming. Therefore, an easy and flexible modeling concept is necessary to support the planer efficiently.
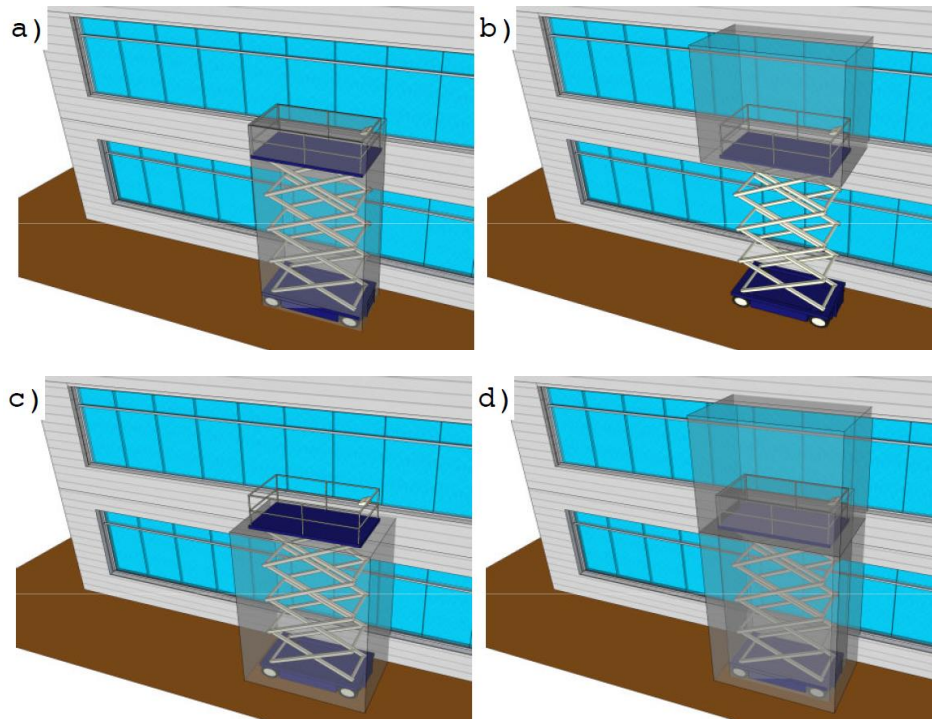
Figure 1: Activity space for installing a window using a hydraulic lift: a) resource space, b) working space, c) safety area, d) cumulative space requirement.

According to Akinci et al. (2000) reference objects are normally used to define spatial requirements. Reference objects define geometric origins at the construction site. All objects modeled in a building information model can serve as reference objects. This includes building elements, storeys, spaces, and service elements. Furthermore, equipment or other movable resources can be considered in case the positions are specified or calculated regarding the execution of related construction activities.

In general, working spaces are modeled by different types of volumes and can be transformed or connected. Therefore, the so-called Spatial Constraint Query Language (SCQL) has been developed. The SCQL is an extension of the Partial Model Query Language (PMQL), which is used to select certain objects of a building information model using the Industry Foundation Classes (Adachi 2003). The PMQL is a XML language to describe operations for selecting, updating, and deleting partial model data and contains recursive object structures and conditional expressions based on SQL. Further information are given in Adachi (2003).

The extensions, which are highlighted in this paper, involve operators to create, transform and connect volumes based on selected reference objects. For the creation of volumes some essential geometries like axis-aligned bounding boxes, object-aligned bound boxes, bounding spheres, bounding cylinders, and bounding cones are defined and implemented. The volumes are created according to the centers and the dimensions of the reference objects. The resulting volume is provided as an IfcSpace object. Consequently, it can be used within further PMQL queries. In Figure 2 the specification of the axis aligned bounding boxes of walls of a certain building storey is highlighted. For each wall a single axis aligned bound box is created. To select the walls of the building storey with the label "Level01" a complex PMQL query with several nested sub queries was used. Extensive knowledge about the entities and structures of the Industry Foundation Classes is required.
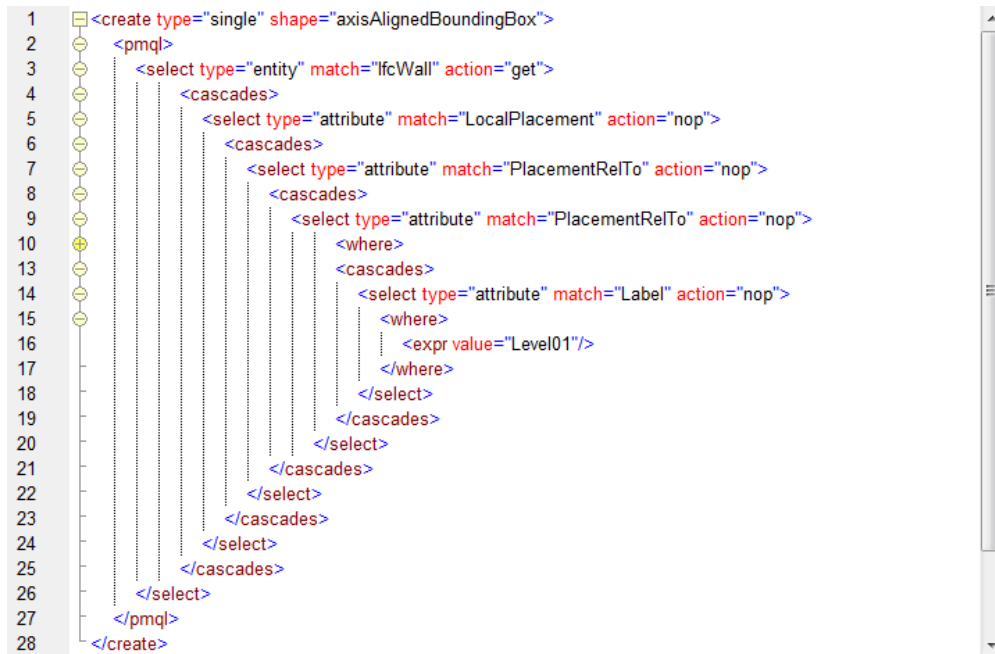
```
 1  <create type="single" shape="axisAlignedBoundingBox">
 2    <pmql>
 3      <select type="entity" match="IfcWall" action="get">
 4        <cascades>
 5          <select type="attribute" match="LocalPlacement" action="nop">
 6            <cascades>
 7              <select type="attribute" match="PlacementRelTo" action="nop">
 8                <cascades>
 9                  <select type="attribute" match="PlacementRelTo" action="nop">
10                    <where>
13                    <cascades>
14                      <select type="attribute" match="Label" action="nop">
15                        <where>
16                          <expr value="Level01"/>
17                        </where>
18                      </select>
19                    </cascades>
20                  </select>
21                </cascades>
22              </select>
23            </cascades>
24          </select>
25        </cascades>
26      </select>
27    </pmql>
28  </create>
```

Figure 2: Creating single working spaces as axis-aligned bounding boxes with reference to walls of a certain storey by using the Spatial Constraint Query Language (SCQL).

Very often dimensions, orientations and positions of created volumes need to be adapted to match spatial requirements. Standard transformation operations like rotation, translation and scaling, or adding an offset are normally used for that purpose. In the context of SCQL these operations have been implemented by additional XML elements. Furthermore, some geometric operations to connect volumes have been defined. Currently, the intersection, union and difference of volumes can be specified to define more complex working spaces.

In Figure 3 the SCQL statement to create the required space for painting a wall in a certain room of a building as spatial requirement is shown. In the first part of the statement an object-aligned bounding box of the wall with a certain global unique identifier is created and a certain offset is specified (cf. Figure 4a). In the second part an object-aligned bounding box of the room, in which the painting should take place, is created. Both volumes are connected by the intersection operator to build the working space for the painting works. The resulting IfcSpace is highlighted in Figure 4b.

```
1   <connect type="intersection">
2     <transform action="offset" operand="(1.2,1.2,1.2)">
3       <create type="single" shape="objectAlignedBoundingBox">
4         <pmql>
5           <select type="entity" match="IfcWall" action="get">
6             <where>
7               <oidis value="0zRmPvpgjCPwW3kEp2EO2C"/>
8             </where>
9           </select>
10        </pmql>
11      </create>
12    </transform>
13    <create type="single" shape="objectAlignedBoundingBox">
14      <pmql>
15        <select type="entity" match="IfcSpace" action="get">
16          <where>
17            <oidis value="1XnXefRHfDj9cnVzdpTOtd"/>
18          </where>
19        </select>
20      </pmql>
21    </create>
22  </connect>
```

Figure 3: Creating the working space for painting works of a wall in a certain room by using the Spatial Constraint Query Language (SCQL).
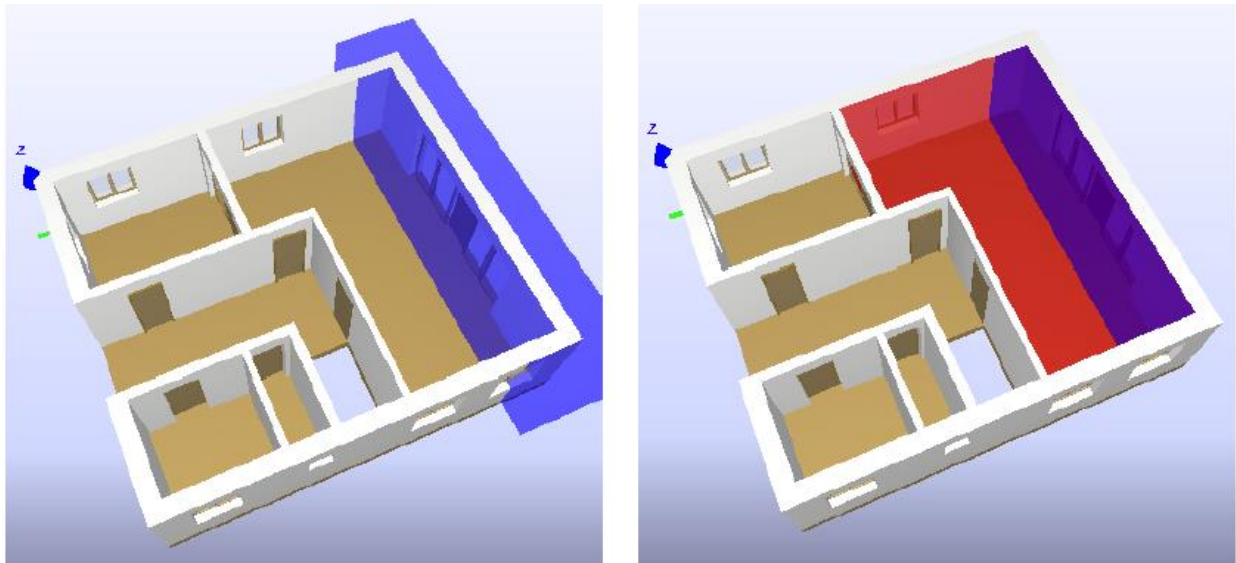
Figure 4: a) Object-aligned bounding box (blue) based on a wall with a certain scale, b) resulting spatial requirement (purple) as intersection with the object-aligned bounding box of a room (red).

The Spatial Constraint Query Language is used to define re-useable spatial requirements based on process patterns. The concept of process patterns is a well-established methodology to support the definition of construction processes. In practice, different construction methods exist to construct a certain building element. For example, the construction of an in-situ concrete wall normally consists of erecting the formwork, installing the reinforcement, placing the in-situ concrete, curing the concrete, and stripping the formwork. These processes can be used for similar building elements. For each process pattern certain attributes can be specified, e.g. performance factors which depend on the bill of quantities, required resources, and needed material (König et al. 2012).

In this paper the process patterns are extended by spatial requirements modeled by means of the SCQL. For each process pattern a set of SCQL statements can be defined and evaluated. As mentioned before, the specification of reference objects is very important to calculate spatial requirements correctly. In the context of construction process definition, process patterns are assigned to certain building elements of a building information model. For example, to instantiate painting processes for a set of walls, the process pattern for painting a wall is assigned to the selected wall objects and rooms where the painting is going to take place (cf. Figure 5). Hence, in this example, each painting process instance is connected to exactly one wall and one room. In consequence, all necessary reference objects are already associated with the single processes and can be reused in terms of defining spatial requirements. For instance, the SCQL statement highlighted in figure 3 can be applied to define a reusable spatial requirement template for the process pattern for painting a wall by deleting the containing *where* clause. In the context of the presented approach, all PMQL select statements without *where* clauses only take into account the selected objects of the building information model. In other words, only building objects are considered which are associated with the process instance based on the applied process pattern. Hence, all reference objects, which could be used in the context of the definition of spatial requirements, have to be assigned to process instances.
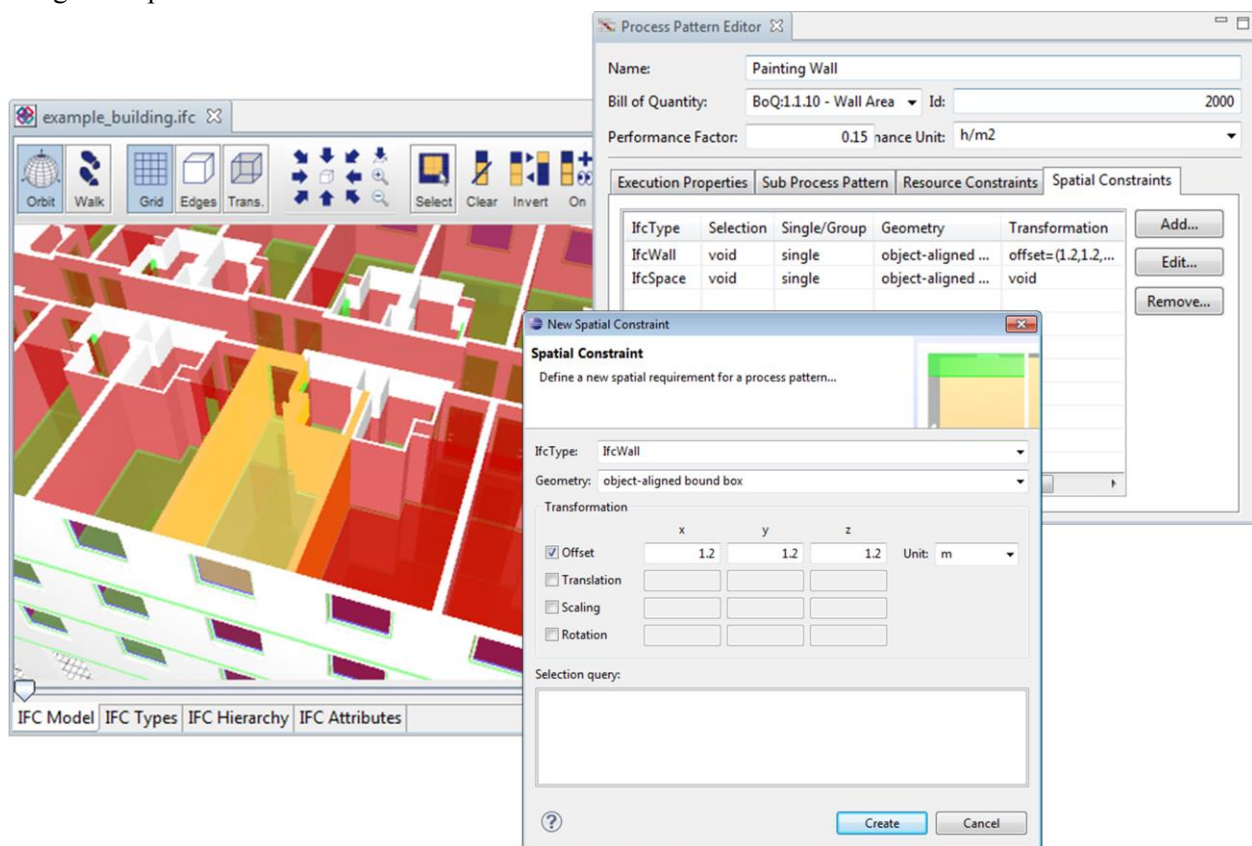

Figure 5: Assignment of the process pattern for painting a wall with spatial requirements to walls of a room of a building storey.

4 SIMULATION-BASED ANALYSIS OF SPATIAL CONSTRAINTS

The analysis of spatial conflicts is performed by applying discrete event simulation. During the simulation the spatial requirements are calculated and spatial conflicts are identified, evaluated and finally solved. Spatial requirements can be integrated into a simulation model like other constraints such as resource or

interdependency constraints. Consequently, spatial requirements must be fulfilled such that a process can be executed. If a spatial requirement is not satisfied, the process cannot be started because of a spatial conflict. The analysis of spatial conflicts depends on all processes which are currently executed. Therefore, the intersection of all working spaces needs to be calculated. Several algorithms exist to calculate collisions between volumes. In this approach hierarchical axis-aligned bounding boxes are used (Gottschalk 2000). The volume of an object is divided into several bounding boxes with different levels of accuracy. Starting with the axis-aligned bounding box of the highest level collision detections are performed. If a collision is detected, the next level will be analyzed. An example of the hierarchical axis-aligned bounding box approach is shown in Figure 6.
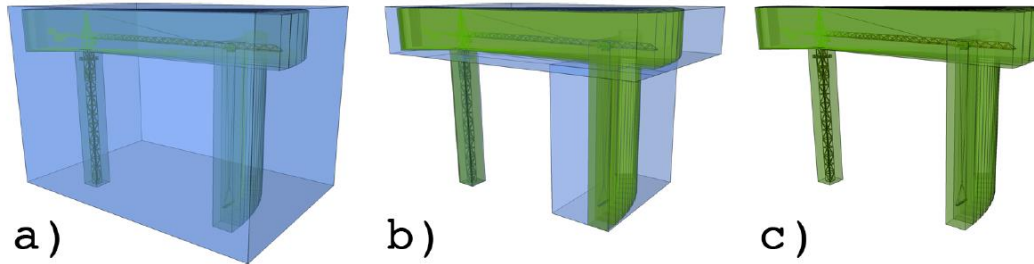

Figure 6: Hierarchical axis-aligned bounding boxes of the safety area during a crane movement a) one bounding box on the highest level, b) finer bounding boxes on a lower level, c) exact approximation of the safety area.

A further concept to reduce the computational effort is the application of octrees for managing the volumes of the working spaces, equipments, buildings and the environment (cf. Figure 7). Depending on the hierarchy of the octree only a few volumes need to be checked using hierarchical axis-aligned bounding boxes. In Mundani et al. (2003) and Borrmann (2007) detailed information about octree algorithms are addressed. As a result, the amount of volumes can be reduced significantly. The octree is updated each time a new working space is occupied by a construction process.
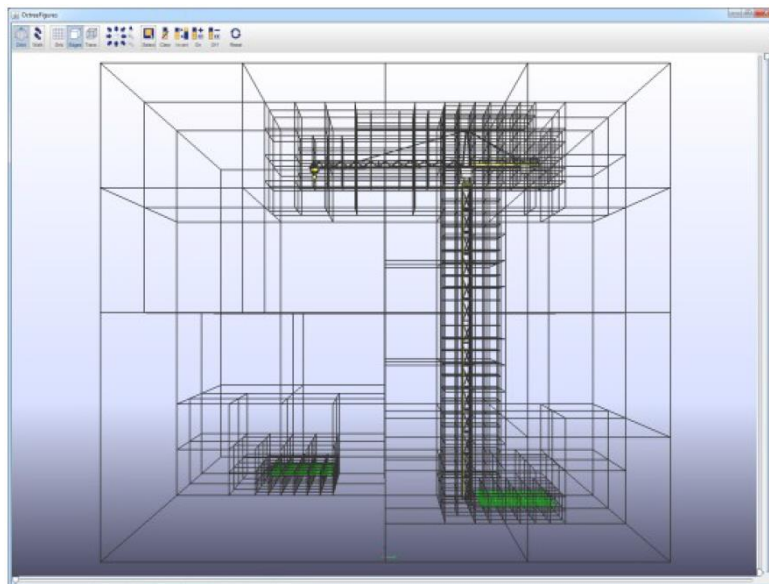

Figure 7: Octree of a tower crane and two storage areas.

For simulating the construction processes a flexible constraint-based simulation concept is used (König et al. 2007). Constraint-based simulation means that each time an event occurs, the constraints are checked to identify which processes can be started next. Generally, hard and soft constraints can be defined and checked. Typically, hard constraints are precedence restrictions or resource requirements. In terms of this paper, spatial constraints are modeled as soft constraint. Soft constraints do not have to be satisfied completely. Rather, a fulfillment degree is calculated to evaluate the satisfaction. The fulfillment degree of a spatial constraint highly depends on the type of the spatial requirement and the dimension of the spatial intersection. For example, for safety zones no collisions are allowed and for working spaces an intersection can be allowed by a low percentage (cf. Figure 8).
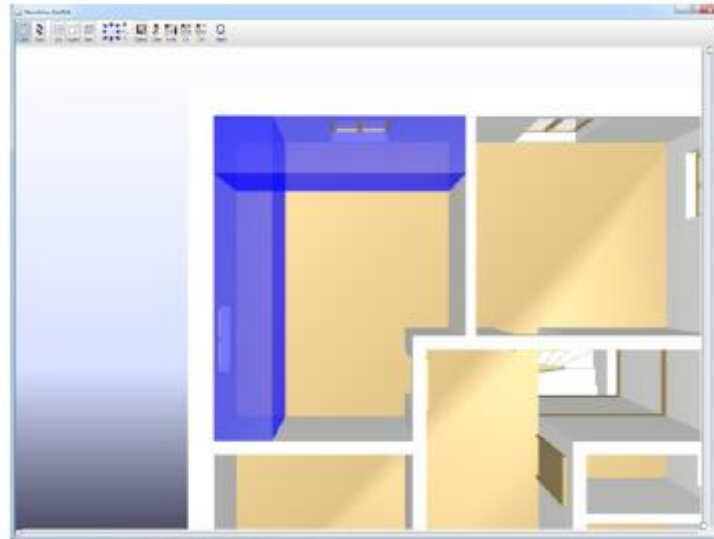

Figure 8: Intersection of the working spaces for painting two walls.

In this approach spatial constraints are modeled by k-weighted constraints. The weights are used to calculate a violation cost factor (Rossi et al. 2006). Additionally, k-weighted constraints have a so-called threshold k that specifies an upper bound for calculating the cost factor. Afterwards, the violation costs can be used to evaluate the spatial conflicts. Another application of the violation costs is the modification of the process duration, other process properties or requirements. For example, for some construction processes the performance needs to be reduced if the intersection percentage lies in a certain interval. The reduction of the performance needs to be considered as long as the spatial conflict is present. That means that the reduction has to be updated each time a process starts or ends. Currently, the reduction factor can only be specified by a linear function of the percentage of intersection. However, more complex algorithms can be integrated easily.

In the following, a simply example using three processes is introduced to demonstrate the calculation of spatial conflicts and their effects on the performance. For better presentation, the spatial requirements are only modeled as axis-aligned rectangles. The processes A and B can be started at the simulation time t0 due to their precedence and resource constraints (cf. Figure 9a).
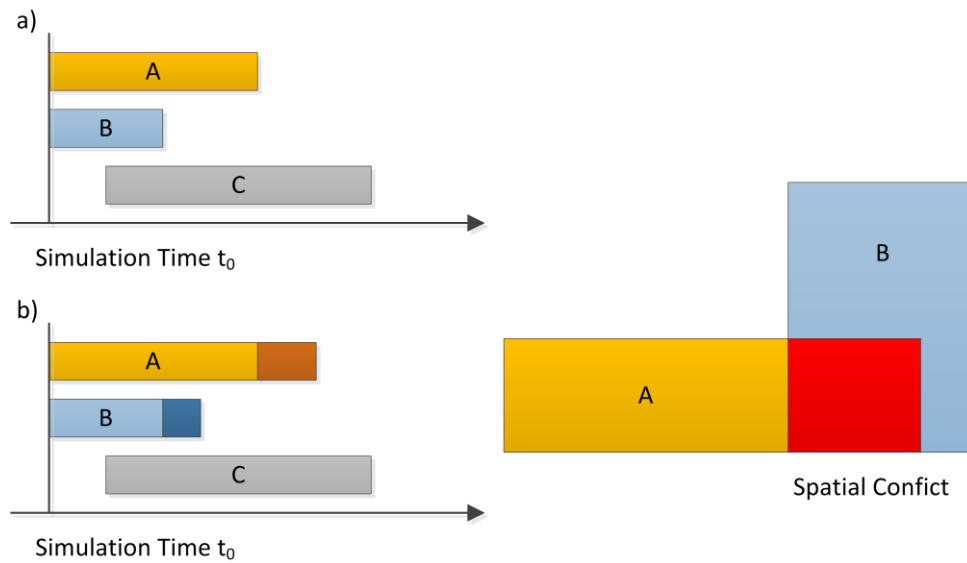
Figure 9: Scheduling processes with a spatial conflict: a) processes A and B with their spatial require-ments and an identified spatial conflict, b) new duration of processes A and B due to the spatial conflict.

In the next step the spatial requirements are generated and checked. An overlapping of 30% and 25 % are calculated for the processes A and B respectively. Both processes can be executed, but the perfor-mance of each process must be reduced by a certain factor. Consequently, updated durations for the pro-cesses A and B need to be calculated (cf. Figure 9b). Now, the processes can be started at simulation time t_0. At the next simulation time step t_1, process C can be executed according to its precedence and resource constraints (cf. Figure 10a). The spatial requirements are calculated and checked again. The working space of process C has a significant intersection with process A by 10%. That means that the durations of process A and C need to be updated based on their reduction factors (cf. Figure 10b). Afterwards, the next simulation time step is performed. The process B is finished and the associated working space is not longer occupied. As a result, the duration of process A needs to be updated in turn.
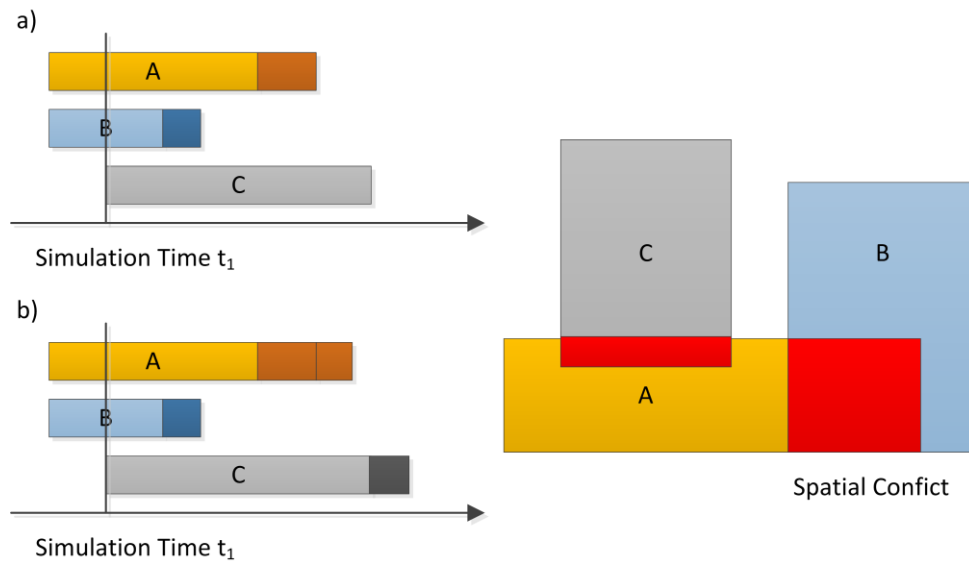
Figure 10: Updating of process durations due to spatial conflicts: a) next executable process C with its spatial requirement and an identified spatial conflict b) new duration of processes A and C.

Modifying and updating process durations as a consequence of spatial conflicts is essential. Therefore, a flexible discrete-event simulation engine is necessary. Available simulation engines do not allow the modification of already scheduled events. In the context of this research a small simulation engine has been developed providing functionalities for updating scheduled processes. Furthermore, the management and calculation of spatial requirements have been integrated.

Another important aspect for making the definition of spatial requirements more practical is the application of well adapted and appropriate software tools. In the last years the so-called SiteSimEditor has been developed at the Ruhr University Bochum to support planers for preparing the necessary input data for simulation-based scheduling. Further information can be found in Marx & König (2011) and König et al. (2012).

5   CONCLUSION AND OUTLOOK

In this paper the modeling and simulation of spatial requirements for construction activities has been presented. Spatial requirements are modeled by using the Spatial Constraint Query Language (SCQL), which is based on the Partial Model Query Language (PMQL). Operators to create and manipulate different geometries using reference objects of a building information model have been introduced. Spatial requirements for construction activities can be defined by integrating SCQL statements into process patterns. For identifying and analyzing spatial conflicts the spatial requirements have been implemented as soft constraints within a constraint-based simulation approach. In consequence, schedules can be generated without any spatial conflict by means of the constraint-based simulation approach. Concepts to update process execution times due to spatial conflicts have been highlighted.

Future research will deal with extensions of the Spatial Constraint Query Language to specify more flexible and complex templates for reusable spatial requirements. Another research topic is the specification of more specific algorithms for schedule updating due to spatial conflicts.

ACKNOWLEDGEMENTS

REFERENCES

Adachi, Y. 2003. "Overview of partial model query language." In Proc. of the 10th Int. Conf. on Concurrent Engineering, 2003, 549-555.

AbouRizk, S. 2010. "Role of Simulation in Construction Engineering and Management." In Journal of Construction Engineering and Management 136 (2010), Nr. 10, 1140-1153.

Akbas, R. 2003. "Geometry-based modeling and simulation of construction processes." PhD thesis, Department of Civil and Environmental Engineering, Stanford University, Stanford, California.

Akinci, B., and Fischer, M. 1998. "Time-Space Conflict Analysis Based on 4D Production Models." In Congress on Computing in Civil Engineering, ASCE, 1998, 342-353.

Akinci, B., Fischer, M., and Kunz, J. 2002. "Automated Generation of Work Spaces Required by Construction Activities." In Journal of Construction Engineering & Management 128 (2002), Nr. 4, 306.

Akinci, B., Fischer, M., Kunz, J., and Levitt, R. 2000. "Formalization and Automation of Time-Space Conflict Analysis. " Stanford University, 06/2000, Nr. WP059.

Borrmann, A. , Schraufstetter, S. , van Treeck, C., and Rank, E. 2007. "An octree-based implementation of directional operators in a 3D Spatial Query Language for Building Information Models." In Proc. of the 24th CIB-W78 Conference on Information Technology in Construction. Maribor, Slovenia.

Chavada, R., Dawood, N., and Kassem, M. 2012 "Construction workspace management: the development and application of a novel nD planning approach and tool. " In Journal of Information Technology in Construction 17 (2012), 213-236.

Choo, H.J., and Tommelein, I.D. 1999. "Space scheduling using flow analysis. In: Proc. 7th Ann. Conf. Intl. Group for Lean Construction, 1999, 299-311.

Gottschalk, S. 2000. "Collision queries using oriented bounding boxes." The University of North Carolina, PhD thesis, 2000.

Kamat, V. R., and Martinez, J. C. 2005. "Large-Scale Dynamic Terrain in Three-Dimensional Construction Process Visualizations." In Journal of Computing in Civil Engineering 19 (2005), Nr. 2, 160-171.

König, M., Beissert, U., Steinhauer, D., and Bargstädt, H.-J. 2007 "Constraint-Based Simulation of Outfitting Processes in Shipbuilding and Civil Engineering." In Proceedings of the 6th EUROSIM Congress on Modeling and Simulation. Ljubljana, Slovenia, 2007

König, M., Habenicht, I., Koch, C., and Spieckermann, S. 2012. "Intelligent BIM-based Construction Scheduling." In Proceedings of the 2012 Winter Simulation Conference.

Lai, K.-C., Kang, S.-C. 2009. "Collision detection strategies for virtual construction simulation." In Automation in Construction 18 (2009), Nr. 6, 724 -736.

Marx, A., and König, M. 2011. "Preparation of Constraints for Construction Simulation." In Proceedings of the 2011 ASCE International Workshop on Computing in Civil Engineering, Miami.

Mawlana, M., Hammad, A., Doriani, A., and Setayeshgar, S. 2012. "Discrete event simulation and 4D modelling for elevated highway reconstruction projects." In Proceedings of the XIVth International Conference on Computing in Civil and Building Engineering, Moscow State University of Civil Engineering.

Mundani, R.-P., Bungartz, H.-J., Rank, E., Romberg, R., and Niggl A. 2003. "Efficient algorithms for octree-based geometric modelling." In Proc. of the Ninth Int. Conf. on Civil and Structural Engineering Computing, Civil-Comp Press, 2003.

Riley, David R., and Sanvido, Victor E. 1997. "Space planning method for multistory building construction." In Journal of Construction Engineering & Management 123 (1997), Nr. 2, 171.

Rossi, F., Van Beek, P., and Walsh, T. 2006. "Handbook of constraint programming." Bd. 2. Elsevier Science, 2006.

Sanders, S.R. 1989. "An analysis of factors affecting labor productivity in masonry construction.", Pennsylvania State University., PhD thesis, 1989.

Tommelein, I.D., Dzeng, R.J., and Zouein, P.P. 1993. "Exchanging layout and schedule data in a real-time distributed environment. In: Computing in Civil and Building Engineering (1993) ASCE, 1993, 947-954.

Zhang, C., Zayed, T.M., Hammad, A., and Wainer, G. 2005. "Representation and analysis of spatial resources in construction simulation." In Proceedings of the 37th conference on Winter simulation Winter Simulation Conference, 2005, 1541-1548.

Zouein, P. P., and Tommelein, I. D. 1993. "Space Schedule Construction." In Computing in Civil and Building Engineering, 1993, 1770-1777.

AUTHOR BIOGRAPHIES

ARNIM MARX received his diploma in civil engineering, with emphasis on computing in engineering, at Ruhr-University Bochum, Germany. Since 2006 he is working as research assistant at the Chair of Computing in Engineering at the Ruhr-University. His research interests are in construction simulation. He is working on a new concept to model and analyze spatial inter-dependencies of resources on construction sites. This concept is the base for a discrete event simulation. Currently he is finishing his PhD thesis, entitled "Considering spatiotemporal constraints during simulation based construction scheduling". His email address is arnim.marx@ruhr-uni-bochum.de.

MARKUS KÖNIG is professor of Computing in Engineering at Ruhr-University Bochum, Germany. Previously, he was assistant professor of Theoretical Methods for Project Management at Bauhaus-University Weimar, Germany. He obtained his Ph.D. in Civil Engineering from Leibniz-University Hanover, Germany in 2003. His research interests include construction simulation, building information modeling, heuristic optimization techniques, knowledge management in construction, intelligent computing in engineering and computational steering. His email address is koenig@inf.bi.rub.de.