GLOBAL PATH PLANNING IN COMPLEX ENVIRONMENTS USING METRIC AND TOPOLOGICAL SCHEMES

Vitaly Semenov, Prof. Dr. / Leading researcher, vital@ispras.ru
OlegTarlapan, PhD / Senior researcher, oleg@ispras.ru
Sergey Morozov, PhD / Senior researcher, serg@ispras.ru
Konstantine Kazakov, PhD student, step@ispras.ru
Vladislav Zolotov, PhD student, step@ispras.ru
Institute for System Programming of the Russian Academy of Sciences (ISP RAS), Moscow, Russia

ABSTRACT

Global path planning is a challenging problem arisen in many fields of research. It is of particular interest to construction planning community facing the requirements of trustworthiness and feasibility of project schedules. Correct schedules must avoid any conflicting situations at project sites and assure the existence of collision-free paths for installed construction elements and deployed equipment. To validate schedules against potential spatio-temporal conflicts, emerging 4D modeling technologies, collision detection and motion planning methods can be applied. Ultimately it would enable detecting and anticipating problems at earlier planning phases and reducing risks and waste at the final construction phases.

Unfortunately, path planning algorithms have relatively high complexity that extremely grows with the input data volume. 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 that makes the stated validation problem highly intractable for construction applications. Being oriented on exact or approximate metric representations, traditional local path planning methods have significant limitations in the case of large-scale environments. Their inability to use overall a priory information on the whole environment creates another shortcoming in global planning. Topological schemas try to overcome these drawbacks by representing the original environment by means of route graphs. Topological schemas scale better than metric ones, but being resistant to geometric representation errors may yield incorrect or suboptimal solutions.

In the paper we propose an effective method leveraging global and local path planning strategies and combining metric and topological schemas. Due to original criteria for extracting a topology from metric information, the method is applicable to complex indoor/outdoor environments and can be used for spatio-temporal validation of construction project schedules. Conducted experiments proved the feasibility and effectiveness of the method presented.

Keywords: 4D modeling, Planning and scheduling, Collision detection, Project validation, Path planning.

1. INTRODUCTION

Global path planning is a challenging problem arisen in many fields of research, including global positioning systems (GPS), autonomous robot navigation, and very large-scale integration design (VLSI) (Dudek & Jenkin 2000). The problem is of particular interest to construction planning community facing the requirements of trustworthiness and feasibility of project schedules (Semenov et al. 2010). Correct schedules must avoid any conflicting situations at project sites and assure the existence of collision-free paths for installed construction elements and deployed equipment. To validate schedules against potential spatio-temporal clashes, emerging 4D modeling technologies, collision detection and motion planning methods can be applied. Ultimately it would enable detecting and anticipating problems at earlier planning phases and reducing risks and waste at the final construction phases.

Recent 4D modeling technologies enable to simulate construction projects ‘in progress’ by consolidating both 3D CAD models and scheduling information delivered from project management systems. Due to immediate visualization the technologies improve spatial reasoning and make communications among project stakeholders more fruitful. Usually, construction projects are visualized as a pseudo-dynamic environment that undergoes known deterministic changes in discrete time moments and remains static between them. Suddenly appeared, disappeared or moved, construction elements and deployed equipment units visually reproduce key project activities.

Popular 4D modeling systems like Synchro, Bentley Schedule Simulator and Autodesk Navisworks support this simulation mode by several reasons. First of all, specification of all the possible paths and kinematic rules the construction objects can move accordingly would take enormous user efforts. Checks against clashes and interferences can be immediately performed under this mode. Finally, simulation of continual behavior of construction objects looks unrealistic for real industrial projects because of consumption of significant computational resources.

Nevertheless, clash and interference checks performed under pseudo-dynamic mode are not comprehensive for spatio-temporal validation of project schedules as they can guarantee the absence of collisions only in discrete time moments rather than over whole time intervals when the objects are moving. Complementary checks against the existence of collision-free paths for moved objects would add the value to the existing validation process. But it requires effective computational methods as the considered mathematical statement is reduced to global path planning problems. Extensive research efforts have been directed toward these problems (LaValle 2006). Unfortunately, path planning algorithms have relatively high complexity that extremely grows with the input data volume. 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 problem highly intractable for construction applications.

Being oriented on exact or approximate metric representations, traditional local path planning methods like configuration spaces, generalized cones, visibility graphs, Voronoi diagrams, probabilistic roadmaps (PRM), rapidly exploring random trees (RRT), potential fields, and cell decompositions have significant limitations in the case of large-scale environments. For details see the comprehensive historical overview (Masehian & Sedighizadeh 2007). Their inability to use overall a priory information on the whole environment creates another shortcoming in global planning. Topological schemas try to overcome these drawbacks by representing the original environment by means of route graphs. Typically, vertices of such graphs are associated with identifiable locations and edges — with possible routes between them. The technical report (Giesbrecht 2004) provides for excellent survey of topological approaches to global path planning. Topological schemas scale better than metric ones, but being resistant to geometric representation errors may yield incorrect or suboptimal solutions (Lamarche 2009).

In the paper we propose an effective method leveraging global and local path planning strategies and combining topological and metric schemas. Topological schema is used for making high-level decisions about perspective routes, metric schema — for local correction of routes and their final validation. Due to original criteria for extracting a topology from available metric information, the method provides for whole coverage of complex indoor/outdoor environments and effectively resolves multiple path planning requests.

The rest of the paper is organized as follows. Section 2 describes peculiarities of adaptive octree structures utilized as a metric occupancy schema. In Section 3 we present an original algorithm and criteria for extracting a topological graph from the metric schema. The algorithms to find suboptimal routes in the topological graph are mentioned in Section 4. The implemented method and results of conducted experiments are discussed in Section 5. In conclusions we summarize benefits of the global path planning method presented and provide recommendations on its practical use for validation of construction project schedules.

2. DYNAMIC OCCUPANCY OCTREE

To simplify the discussed motion planning problem and to avoid computationally expensive analysis of whole 3D environments, so-called space (or cell) decomposition methods are usually applied. They assume subdividing the environment space into cells and determining occupancy status for each localized cell. Using the occupancy metric schema, path planning problem can be solved more efficiently by successive finding neighboring free cells and navigating over them from a starting point of the moved object to its destination point. Importantly, it can be done under very general assumptions about the simulated environment.

2.1. Environment specification

We suggest the environment is composed of objects which may be geometric 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). No matter which geometric models are adopted. It is assumed only that there is a common function for the interference identification between any environmental object and any given box. The testing result is an occupancy status taking the values 'grey', 'black' or 'white' and pointing whether the box is partially occupied, entirely full, or entirely empty correspondingly.

The behavior of all the environmental objects is completely known and it corresponds to the pseudo-dynamic simulation mode defined above. According to this mode, the environment can be considered as static except of a finite set of discrete time moments when new objects may suddenly appear and existing objects may disappear or move to new locations. It is suggested that at any time moment only one object may change own state and the environment in whole. In such approach, the only active object is considered as movable to be subjected to path planning and other passive objects — as the environment obstacles. Next time moment, the objects may exchange roles so that another object may become active as the previously active object is interpreted as a passive obstacle.

As opposed to many works addressing to path planning, no specific restrictions are imposed upon the geometric representation of both simulated environment and moved objects considered as its intrinsic parts. It is worth to mention the works (Thrun 2008, Fabrizi & Saffiotti 2000) where space decomposition algorithms are proposed for static environments and robots which shape is restricted to a circular cross-section and primitives like spheres, cylinders, polygons, or points. These examples show that space decomposition approach has limitations for dynamic environments and movable objects of non-trivial shapes. Nevertheless, no principal restrictions are done for the simulated environments in our approach.

2.2. Adaptive cell decomposition

We preferred to use adaptive cell decomposition and dynamic octree structures for metric representation of the simulated 3D environment. Being adapted to many computer graphics applications such as frustum culling, occlusions, collision detection, they are also used for motion planning by offering the capability of hierarchical multi-level representation of the environment and spatial reasoning when navigating over it.

Adaptive cell decomposition is used to reduce the number of cells suffered to the analysis and to waste less memory storage space and computation time. Contrary to regular cell decomposition, it is a good tactic for the complex indoor/outdoor environments simulating construction sites and containing large regions with the same traversability. Adaptive decomposition relies on the fact that much of the information in the free space is redundant in regular cell decomposition. The regular shape of the cells is maintained, but the cells are recursively reduced in size in order to both use the space more efficiently and maintain as much details as possible.

Unfortunately, adaptive cell decomposition imposes problems for dynamic and pseudo-dynamic environments. When changes are happened, it is necessary for the entire data structure of the metric schema to be completely revamped or to be correctly updated. In addition, irregular cell decomposition has difficulty in providing near optimal paths and often results in jagged paths. One good solution is the use of framed trees as suggested in (Szczerba & Chen & Uhran 1998). However, in high clutter environments framed structures can be less efficient than regular grids, due to the overhead required to keep track of the cell sizes and locations.

2.3. Occupancy octree generation

To generate the occupancy octree the well-known adaptive decomposition method is applied. It begins by imposing a large size cell over the entire planning space. If a grid cell is partially occupied, it is sub-divided into eight equal subparts or octants, which are then reapplied to the planning space. These octants are then recursively subdivided again and again until each of the cells is either entirely full or entirely empty. The subdivision process is interrupted also for refined cells if their size becomes equal or smaller than a given tolerance of the generated metric representation. The resulting octree has grid cells of varying size and concentration, but the cell boundaries coincide very closely with the obstacle boundaries. An example of the octree representation is shown in Figure 1b. It was generated for a simple building model presented in Figure 1a.

The cells of the deployed octree are then marked as 'grey', 'black' or 'white' depending on their occupancy status that points out whether the cell is partially occupied, entirely full, or entirely empty. Note that sometimes it is difficult to identify entirely full cells for the environments consisted of, so-called, polygon soups rather than solid primitives or assemblies. In such situations the subdivision process has to be recursively continued under suggestion that the cells are partially occupied regions. By construction, white and black cells are always leaves, but grey cells may be intermediate octants too. Grey leaf cells have the size equal to the given tolerance. Other cells may have larger sizes.

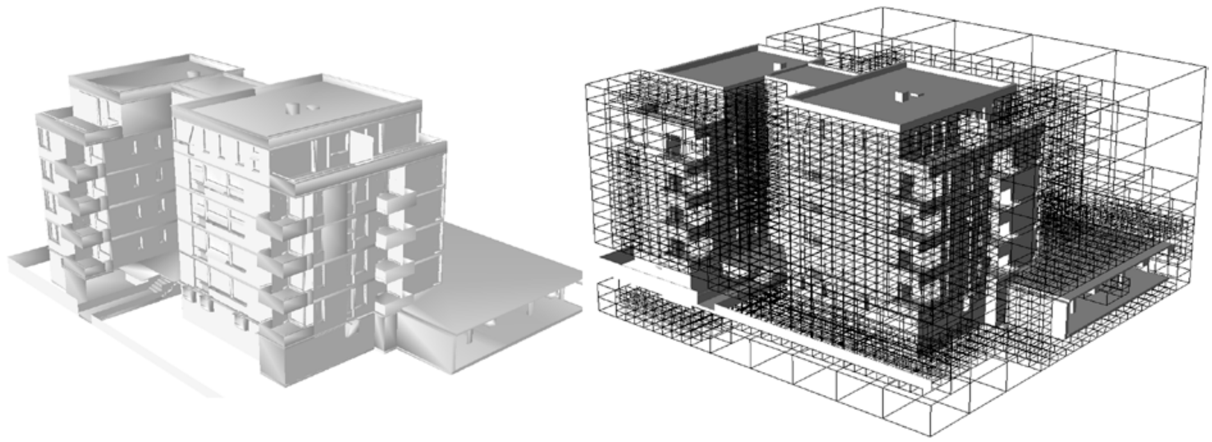

Figure 1: (a) Geometric representation of the building, (b) Deployed occupancy octree.

2.4. Computing distance field

To make the metric representation more constructive for spatial reasoning and topology extracting, it is proposed to enrich the octree structure by distance field values. For this purpose empty cells store distances to the nearby obstacles. Euclidian distance provides high estimate for the radius of a solid ball which can be safely placed in the center of the cell without any risk to collide with the environment obstacles. To simplify computations, the distance estimates can be obtained using already deployed metric schema and avoiding consumable analysis of the original geometric representation. In the proposed algorithm distances are computed from the center of each empty cell to the nearest faces, edges or corners of the cells (entirely or partially) occupied by the environment objects. The algorithm tends to minimize the number of cells suffered to the analysis. It runs by traversing all empty octants, collecting neighboring cells for each such octant and determining the distance to nearest occupied cell or cells.

Algorithms for finding neighbors in octree structures are well-known (Kim & Lee 2009). Basically, they are capable to collect the direct neighbors defined by those cells the faces, edges or corners of which are directly adjacent to the explored octant. These are classical neighbor finding method, backtracking method, matrix based strategy using locational binary code, finding method in each 26 directions, address encoding technique.

The algorithm applied works in a way similar to the backtracking method mentioned above. Neighboring cells are determined by finding common ancestor and by descending from it to adjacent cells of the explored octant. A principal distinction of the presented algorithm is that it collects also indirect neighbors which are located not far than search radius r from the explored octant. It enables to avoid redundant analysis of remote cells and to minimize total computations if some occupied neighbors have been already found. An initial estimation for the search radius is given by the expression $\sqrt{3}(3/2\, a - d)$, where a is the linear size of the explored octant and d is the size of the smallest cell coincident with the given metric tolerance. Indeed, by construction the parent of the explored empty octant may be only a grey cell and it must contain at least one occupied child cell. Evidently that it is located not far than $\sqrt{3}\,(a-d)$ from the center of the parent cell and not far than $\sqrt{3}(\frac{3}{2}a-d)$ from the center of the explored octant. As new neighbors are found and the distance value is decreased, the estimated search radius r can be refined and replaced by the current value of the distance. This leads to additional savings on traversing and processing cells. An estimated search area and an occupied cell nearby to the central empty octant are shown in Figure 2a as a circle and a bold arrow correspondingly. Figure 2b illustrates the distance field distribution computed for the building model and the occupancy octree presented above.

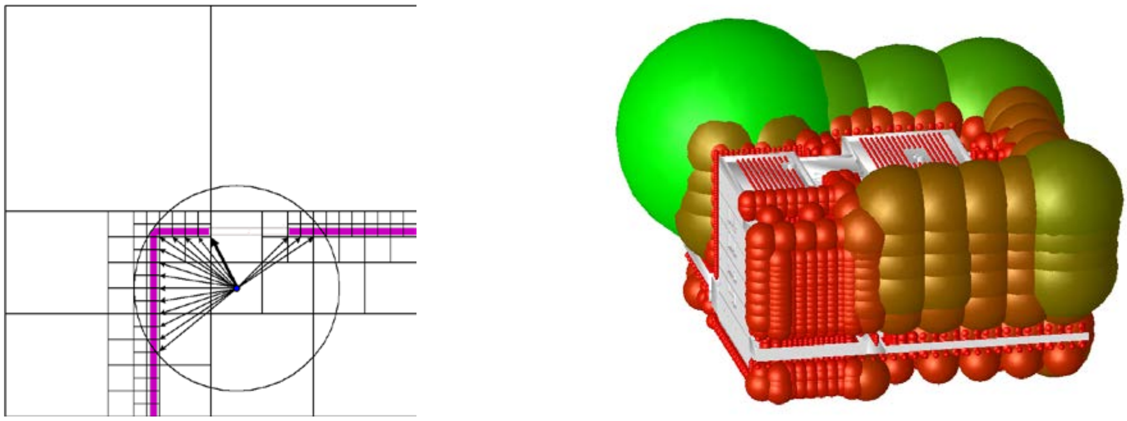

Figure 2: (a) Estimated search area and neighboring occupied cell, (b) Distance field distribution.

3. EXTRACTING TOPOLOGICAL SCHEMA

As explained, the occupancy octree is an useful metric representation to perform path planning in 3D environments by successive navigating over empty octants and avoiding the environment obstacles. However, being represented by huge number of cells the octree prevents efficient coverage in large-scale environments. Usually indoor environments consist of relatively small rooms sequentially connected via doors, corridors, stair wells, so room-to-room coverage path planning looks more efficient than cell-to-cell navigation. This observation inspired us to follow “space-gate” reasoning paradigm and to support corresponding topological schema that represents extracted spaces and gates of the environment, possible passages and perspective waypoints for further global path planning. Outdoor environments can be also thought as spaces connected via gates if they are mentally enclosed by some surrounding borders of the project site. Both spaces and gates are considered as non-overlapping, simply connected subsets of empty cells of the occupancy octree. Principal difference between introduced categories is that the spaces approximate large free regions of the environment, whereas the gates — small narrow regions. In the next subsection these categories are defined more exactly taking into account the distance field distribution.

Therefore, the objective of this study is to extract a topological schema from available metric information. This problem attracted many researches which tried to obtain adequate topological schema by identifying features like virtual doors, narrow passages, corners, middle lines (Myung et al. 2009, Choi et al. 2009). Although their methods work well for particular domestic cases, scarcely that they match to complex environments simulating real construction projects in progress. The proposed algorithm for extracting a “space-gate” topological schema looks very promising from this point of view.

3.1. Space and gate identification

As explained above, the Euclidian distances can be obtained for each empty cell of the octree by traversing its neighbors and determining minimal distance from a center of the cell to nearest faces, edges or corners of the occupied neighbors. Thus approximated distance field can be utilized for extracting free regions of the environment and identifying them as spaces or gates.

Underlying principle for detecting free regions is to find the cells where the distance field reaches local maxima. Each such maximum originates a subset of empty, simply connected cells surrounding it and having distance values not exceeding the local maximum. Opposite to centerline algorithms oriented on rectangular occupancy grids and steepest descent by gradient vector approximations, our algorithm is applicable to arbitrary irregular grids and octrees which of special value for the discussed global path planning problems.

To explain how the algorithm works, let us define binary relations of the adjacency, reachability, dominancy and origination among empty cells of the occupancy octree C. The cells $c', c'' \in C$ are adjacent (or $c' \sim c''$) if and only if there is a common face belonging to boundaries of both cells. The cells $c', c'' \in C$ are reachable one from other ($c' \leftrightarrow c''$) if there is a sequence of the cells $c_1, c_2, ..., c_n \in C$ so that $c' \sim c_1, c_1 \sim c_2, ..., c_n \sim c''$. Obviously that the reachability is obtained by transitive closure of the symmetric adjacency relation on the set C. The cell $c' \in C$ dominates over the cell $c'' \in C$ (or $c' > c''$) if and only if the cells are adjacent and estimated distance value for the cell c' is larger than the corresponding value for the cell c''. And, finally, the subset $C' \subseteq C$ is originated from the cell $c' \in C'$ ($c' \triangleright C'$) if and only if for any $c'' \in C'$ there is a sequence $c_1, c_2, ..., c_n \in C'$ so that $c' > c_1, c_1 > c_2, ..., c_n > c''$ and there is no dominating cell $c''' \in C'$ for c' so that $c''' > c'$. The introduced adjacency and reachability relations are symmetric, reflexive and transitive. The dominancy relation is transitive.

We define the regions to be subsets of cells $R_1, R_2, R_3, ... \subseteq C$ obtained by transitive closure of the dominancy relation on a set of all empty cells of the octree C so that $R_1 \cup R_2 \cup R_3 \cup ... = C$ and exactly one originating cell is associated with each detected region $s_1 \triangleright R_1$, $s_2 \triangleright R_2$, $s_3 \triangleright R_3$. Then the spaces $S_1, S_2, S_3, ... \subseteq C$ can be defined as non-intersecting subsets of regions $S_1 = R_1 \setminus \{R_2 \cup R_3 \cup ...\}$, $S_2 = R_2 \setminus \{R_1 \cup R_3 \cup ...\}$, $S_3 = R_3 \setminus \{R_1 \cup R_2 \cup ...\}$ with the same originating cells $s_1 \triangleright S_1$, $s_2 \triangleright S_2$, $s_3 \triangleright S_3$. Consider now a remained subset $G = \{R_1 \cap R_2\} \cup \{R_1 \cap R_3\} \cup \{R_2 \cap R_3\} ... \subseteq C$ representing all the intersected regions with more than one originating cell. Cells of the subset G with the same combination of originating cells are grouped to form the gates $G_1, G_2, G_3, ... \subseteq G \subset C$. Transitive closure of the dominance relation is applied on each gate set and if more than one originating cell is found, the gate is subdivided to have strong correspondence $g_1 \triangleright G_1$, $g_2 \triangleright G_2$, $g_3 \triangleright G_3$. One would assume that the extracted spaces and gates have no mutual intersections $C_i \cap C_j = \varnothing$, $G_k \cap G_l = \varnothing$, $C_i \cap G_k = \varnothing$, $i \neq j$, $k \neq l$ and they form an exact cover of the original set $S_1 \cup S_2 \cup ... \cup G_1 \cup G_2 \cup ... = C$.

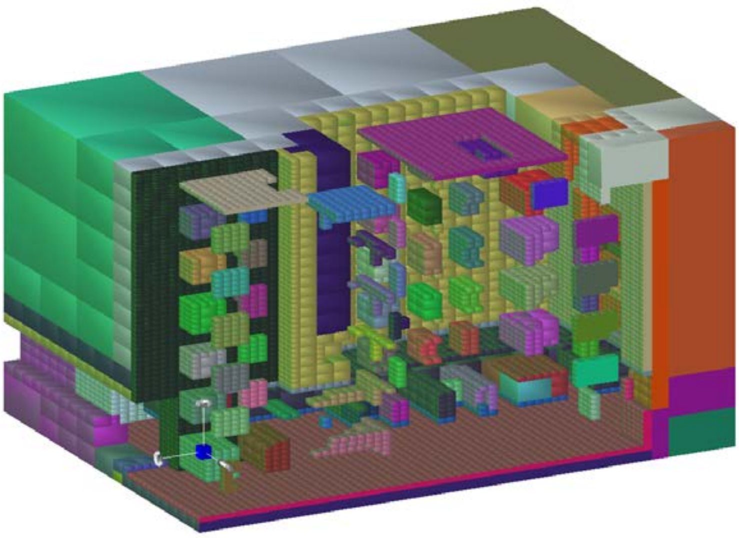

Figure 3: Space and gate regions identified in the occupancy octree.

The introduced definitions and obtained formula give a constructive algorithm to identify spaces and gates and to extract a topological schema from the metric representation. As explained above, the spaces are originated from the cells in which the distance field reaches local maxima. Therefore, the spaces consolidate relatively large and wide free regions. It could be suggested that the gates can be formed by analyzing local minima of the distance field, but it is not true as it takes zero values exactly on the obstacle boundaries. In our algorithm the gates are derived as intersections of regions, therefore they correspond to relatively small and narrow passages connecting the dominating spaces. An important advantage of the algorithm is an avoidance of any domestic feature analysis and its applicability to both indoor and outdoor environments. An example of the space and gate identification is shown in Figure 3.

3.2. Obtaining topological schema

According to the algorithm presented, exactly one originating cell is associated with each space or gate. The centers of such cells should be good initial approximation for the waypoints on paths crossing corresponding spaces and gates. Indeed, the computed centers never leave free region of the environment, tend to remain in the "middle" being situated as far from the obstacle boundaries as possible. This makes the choice of such waypoints quite motivated.

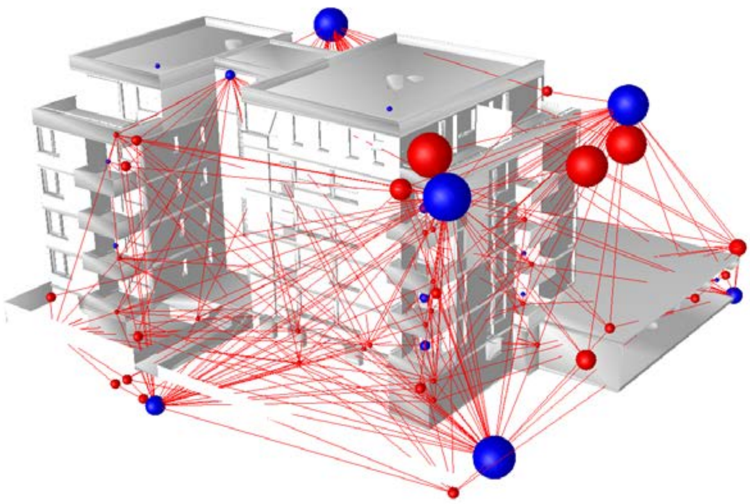

Figure 4: A bipartite "space-gate" topological graph.

All the recognized spaces and gates match to corresponding vertices of the generated bipartite topological graph. Space vertices are connected by edges with incident gate vertices, but not with other space vertices. Similarly, gate vertices are connected with incident space vertices, rather than with other gate vertices. Thus, spaces and gates are alternated when navigating over environment and traversing the topological graph. A bipartite topological graph generated for the presented building model is shown in Figure 4.

The additional metric information is assigned to vertices and edges of the topological graph. Each vertex includes the information about space or gate it matches, coordinates of its originating cell and the distance value. Each edge represents how incident space and gate vertices are connected and stores a length of the direct line between centers of their originating cells.

4. ROUTE PLANNING

4.1. Search in topological schema

The observations above give a general method how global path planning can be performed using this topological schema. First of all, source and destination points are localized in both the deployed metric representation and the extracted topological graph. Then optimal or suboptimal route is found by well-known graph search algorithms taking into consideration some criterion and strongly specified cost function.

Classical Dijkstra's algorithm solves the single-source shortest path problem for arbitrary directed graphs with unbounded nonnegative weights. The algorithm finds the path with lowest cost between a given source vertex and every other vertex of the graph. Once the shortest path to the single destination vertex has been determined, the algorithm stops. It is worth to mention a version by Fredman and Tarjan that is asymptotically faster than the original algorithm. Note that many other graph search algorithms like depth-first, breadth-first, iterative deepening, uniform-cost, best-first, A* search can be applied for finding suboptimal routes in shorter time. For a more complete description see the survey (Norvig & Russell 1995).

In our method the classical Dijkstra's algorithm is used with varied cost functions corresponding to different target criteria, e.g. to find shortest route or most reliable route passing through large free regions and rising up the chances to safely move the object.

The first statement implies that edge weights are simple lengths between spaces and gates (or more exactly, between their originating cells). Under such interpretation, the algorithm finds metrically the shortest route between source and destination vertices. The spatial path can be easy formed from waypoints lying in the centers of incident spaces and gates associated with the found route. Note that the graph-search algorithm cannot guarantee that the found route will be a collision-free path for particular moved objects. But it can be a good initial approximation for further correction by a local planning algorithm exploiting all the details of their geometric representations.

The second statement focuses on search of more reliable, trustworthy routes. The cost function is defined as a minimum of the distance values for all spaces and gates the route crosses. Then the algorithm gives a route promising for moving large balls. It is achieved due to selection of those spaces and gates which have maximum values of the Euclidian distance to nearby obstacles and, therefore, are most suitable for passing ball-shaped objects. For other geometric objects the algorithm may give imperfect solutions and local path planning is necessary to correct the intermediate results.

4.2. Combining global and local path planners

As explained above, search in a topological graph can be categorized as a global planning strategy relying mostly on a priori information on the environment and assuming its whole coverage. The main weakness of global strategies is that they usually plan routes by ignoring geometry features of environment objects, therefore found solutions sometimes turns out to be useless. It seems to be a right idea to use a global algorithm only on a rough, larger scale, topological schema and to apply a local algorithm on a detailed metric or geometric representation of the moved object and its immediate surroundings. It looks just as humans do when determining how to get from one point to another in a building: initially they only decide which rooms and doors to pass through and then they move carefully bypassing corners and small obstacles. Therefore, the principle of combining global and local planners fits our purposes best because none of the strategies, being applied separately or discordantly, gives the good result.

Using the "space-gate" topological schema and graph search algorithms, we can determine at any moment what the possible path to get to a certain destination is. Then the path is checked against potential collisions and, if necessary, it is corrected using popular local planning algorithms like visibility graphs, Voronoi diagrams, probabilistic roadmaps (PRM), rapidly exploring random trees (RRT), and potential fields. For more details see the monograph (Lavalle 2006).

We prefer a known modification of the RRT algorithm that assumes growing trees from neighboring waypoints in opposite directions. For examples, see randomized kinodynamic planning. In such way, the waypoints are linked and the resulting path is formed as a collision-free polyline close to the original path. It can be suffered to additional post-processing to reduce the number of waypoints and to make the path more convenient for practical navigation purposes. The algorithm parameters like the penetration step and the allowable size of expanded trees help to control the accuracy and to interrupt the process if the initial approximation turned up imperfect and the final result cannot be obtained in reasonable time. In such critical situations the alternative routes must be determined by global strategies and be carefully explored by local strategies to complete problem solving. Figure 5 provides an example of the resulting path in the presented building model obtained using both global and local planning strategies.

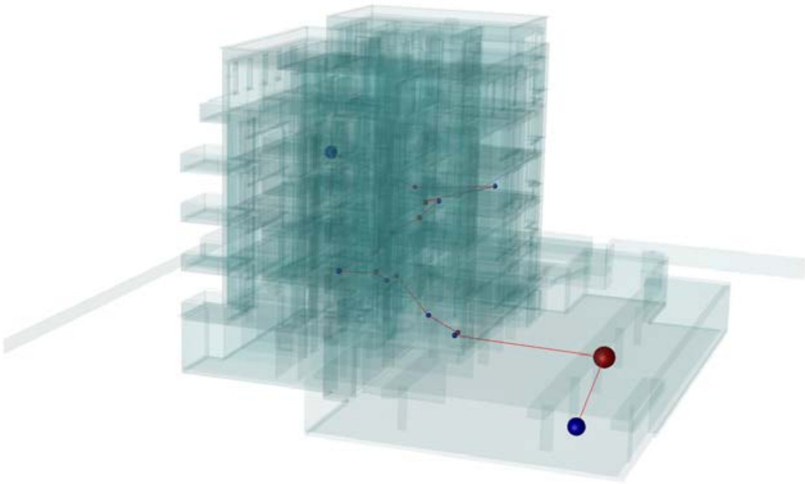

Figure 5: A resulting path obtained using both global and local strategies.

4.3. Renewal of metric representation and topological schema

As known the metric representations and occupancy octrees have difficulties with dynamic environments being underwent to changes. If the changes happened, the octree structures must be updated accordingly to be useful for further spatial analysis. For arbitrary translated and rotated objects there are fast algorithms to update occupancy octree using object grids and applying transformations directly to them. Nevertheless, the algorithms become useless for the discussed pseudo-dynamic environments where the objects may appear and disappear without any continual motions. Moreover, a problem of updating the extracted topological schema remains open too. For these purposes more general principles and more advanced reference architecture must be developed.

More attractive solution is to store cross-references among objects of the original environment, deployed octree cells, extracted space and gate regions, and vertices of the final topological graph. If any object underwent some changes, the octants occupied by the object in its previous and new positions were localized and analyzed against the necessary updates. Due to the maintained references such updates remain local and can be done in reasonable time without traversing the whole octree structures. The cells occupied by the object in the new position hold or change the status to grey or black. Subdivision process must be initiated to get adequate metric representation of the changed object surroundings. The cells occupied by the object in the old position change the status to white on the condition that no other objects are placed in the cells. In this case parent cells must be recursively rechecked and collapsed if they become empty. In order to update distance values, the references among empty cells and nearby objects are supported too. It makes the recalculation of distances efficient, because of a small number of referenced empty cells has to be suffered to the additional analysis.

Similar principles are utilized for referencing and updating the topological graph. If the distance field was locally changed in some cells, corresponding space and gate regions are recalculated and the topological graph is restructured. Its metric attributes like centers of originating cells and distances are updated too. For brevity we omit here the details how a cascade of the updates is carried out avoiding redundant actions. Thus, the introduced reference architecture enables to effectively update and to maintain all the deployed metric and topological schemas in consistent state under all the typical events appeared in pseudo-dynamic environments.

5. SOME EXPERIMENTAL RESULTS

To validate the global path planning method proposed and to estimate its practical benefits, we have implemented a program and conducted timing experiments, in which a middle-size building model was applied. Although the model was strongly structured in accordance with IFC standard, in fact a polygon soup consisted from about 500,000 triangles was taken as original 3D geometry data. The tolerance of the metric representation was limited by the regular grid sizes 52 x 56 x 30.

The first series of the experiments corresponded to static simulation mode assuming global path planning in the constructed building. The conducted analysis detected 112 spaces, 62 gates and 354 edges. The analysis took totally 107 CPU seconds on a typical computer configuration Core 2 Duo E8600 processor (2.13 GHz), 2GB of RAM (800 MHz). After the topological graph has been constructed, search in the graph and final validation of the found spatial routes took no more than a few seconds that looks like quite promising result giving an opportunity to interactively resolve similar problems.

The second experiment series reproduced pseudo-dynamic simulation mode assuming the validation of a construction plan for the same building model. The plan was scheduled so that all the construction elements to be installed consequentially each other without any continual motions. For every installed element the existence of collision-free paths was proved. On the specified computer configuration full validation of the construction project schedule took about 7 CPU minutes that illustrates significant performance of the method and its applicability to the discussed computationally hard problems.

6. CONCLUSIONS

Thus, the effective path planning method has been proposed. Leveraging global and local strategies and combining metric and topological schemas, it is applicable to complex large-scale environments. Conducted experiments proved the feasibility and effectiveness of the method in industry valuable statements. Next efforts will be devoted to spatio-temporary validation of project schedules and to realization of specific checks against existence of collision-free paths for installed construction elements and deployed equipment units. These activities are planned as a further improvement of emerging 4D modeling and planning technologies.

REFERENCES

Choi J. W., Choi M. Y. and Chung W. K. (2009) “Incremental Topological Modeling using Sonar Gridmap in Home Environment.” *In International Conference on Intelligent Robots and Systems*, 3582-3587.

Dudek G. and Jenkin M. (2000) “Computational Principles of Mobile Robotics.” Cambridge University Press, Cambridge, UK.

Fabrizi E. and Saffiotti A. (2000) “Extracting Topology-Based Maps from Gridmaps.” *In International Conference on Robotics and Automation*, 2972-2978.

Giesbrecht J. (2004), “Global Path Planning for Unmanned Ground Vehicles”, *In DRDC Suffield TM 2004-272*, Defence R&D Canada – Suffield, 1–41.

Kim J., Lee S. (2009) “Fast neighbor cells finding method for multiple octree representation” *In IEEE Int. Symposium on Computational Intelligence in Robotics and Automation.* 540-545.

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

Lavalle S. M. (2006) “Planning algorithms.” Cambridge University Press, Cambridge, UK.

Masehian E., Sedighizadeh D. (2007) “Classic and heuristic approaches in robot motion planning – A chronological review”, *In Proceedings of world academy of Science, Engineering and Technology*, 101–106.

Myung H., Jeon H. M., Jeong W. Y. and Bang S. W. (2009) “Virtual Door-Based Coverage Path Planning for Mobile Robot.” *In Federation of International Robot-soccer Association*, 197-207.

Norvig P. and Russell S. (1995) “Artificial Intelligence: A Modern Approach.” Prentice Hall.

Semenov V.A., Kazakov K.A., Morozov S.V., Tarlapan O.A., Zolotov V.A., Dengenis T. (2010) “4D modeling of large industrial projects using spatio-temporal decomposition.” *In eWork and eBusiness in Architecture, Engineering and Construction, eds. K. Menzel and R. Scherer*, 89-95.

Szczerba R.J., Chen D.Z., Uhran J.J. (1998) “Planning Shortest Paths among 2D and 3D Weighted Regions Using Framed-Subspaces” *International Journal of Robotics Research.* 17(5). 531-546.

Thrun S. (1998) “Leaning Metric-Topological Maps for Indoor Mobile Robot Navigation.” *Artificial Intelligence*. 99. 21-71.