Hierarchical Voronoi-based Route Graph Representations for Planning, Spatial Reasoning, and Communication

Jan Oliver Wallgrün[1]

Abstract. In this paper we propose a spatial representation approach for a mobile robot operating in an office-like indoor environment which is intended to provide an interface between low-level information required for navigation and abstract information required for high-level symbolic reasoning about routes. The representation is based on a route graph [16] that links navigational decision points via edges corresponding to route segments. We describe a particular route graph representation that is derived from the generalized Voronoi diagram of the environment and enables the robot to incrementally construct the representation autonomously. Since the Voronoi-based route graph still reflects irrelevant features of the environment, our proposed representation is a hierarchical structure consisting of route graph layers representing the environment at different levels of granularity. It is shown how the more abstract layers can be derived from the original route graph by using relevance measures to assess the significance of the vertices. We provide examples of how planning, spatial reasoning, and communication can benefit from this kind of representation.

1 Introduction

In the context of mobile robot control systems, a crucial step is to decide in which way spatial information about the robot's environment required to solve different subtasks like path planning, path execution, localization, spatial reasoning, etc. should be stored. Since any truly autonomous mobile robot will have to be able to construct and maintain its model of the environment on its own based on observations, the representation will have to bridge the gap from low-level sensor data to entities needed for high-level reasoning.

Representations used in current mobile robot systems can be grouped into two main classes: Metric approaches [12, 10, 14] rely on an absolute coordinate system superimposed onto the environment to specify position and orientation of spatial entities. Topological representations on the other hand represent the environment by a graph structure that explicitly stores spatial relations like adjacency or connectivity between the entities represented by the vertices [7, 3].

In this paper we describe a so-called route graph representation (a concept introduced in [16]) which is a special kind of topological map in which the graph structure describes qualitatively different routes through the environment. The vertices in this representation correspond to navigational decision points, while the edges correspond to route segments connecting the decision points. The representation and the involved procedures are intended to serve as a navigation and mapping module within a hybrid control architecture combining reactive and deliberative components.

Our particular route graph representation is based on the generalized Voronoi diagram (GVD) which is a retraction of the free parts of the working space onto a network of one-dimensional curves reflecting the connectivity of free space (see figure 1). The GVD allows us to derive a route network from information about the obstacle boundaries. The graph corresponding to the GVD, the generalized Voronoi graph (GVG), forms the core of our route graph representation and is annotated with additional information like relative positions of the vertices. While the resulting route graph can be used for navigation by applying a simple motion behavior to travel along edges until the next vertex is reached [3], it still contains parts that are caused by insignificant features of the environment like small niches or that result from noise in the sensor data and that are irrelevant for high-level reasoning. Therefore, we develop a way to deal with this problem by deriving more abstract route graphs from the original GVG employing measures to assess the relevance of the Voronoi vertices and the regions accessible by them. Based on the ability to abstract from the GVG, we propose a hierarchical organized multi-layer representation with the original GVG at the bottom level and layers containing route graphs representing the environment at different levels of granularity stacked on top of it. Corresponding features in adjacent layers are linked with each other allowing to switch to a finer or coarser level of granularity. We argue in favor of such a representation for a mobile robot system for application in office-like indoor scenarios showing how it is particularly well-suited to provide an interface between low-level navigational information and abstract information required for high-level planning, reasoning, and communication. Planning and spatial reasoning based on this representation can be performed in a hierarchical manner to make them more efficient.

The paper is structured as follows: Section 2 describes the Voronoi-based route graph representation scheme and briefly discusses advantages of the representation and important issues like incremental construction and localization. In section 3 the idea of a hierarchization of the Voronoi-based route graph is elaborated and relevance measures are proposed to derive such a representation from the original route graph. Section 4 presents planning and reasoning examples within this kind of representation and section 5 provides first results of the experimental evaluation of the described approach.

2 Voronoi-based route graph representation

The GVD is a generalization of standard Voronoi diagrams [1] that handles other geometric primitives, e. g. line segments [9, 6], instead of only point sites. It is also related to the idea of a shape's skeleton

[1] Department of Mathematics and Informatics, Universität Bremen, Germany
email: wallgruen@informatik.uni-bremen.de

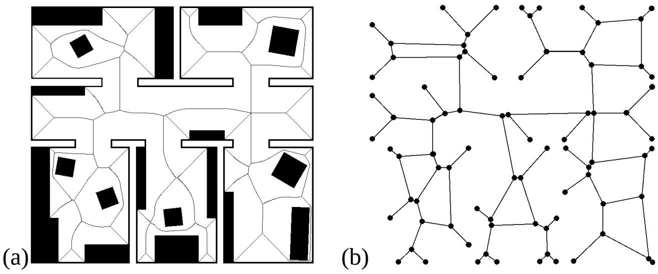

Figure 1. (a) The generalized Voronoi diagram (GVD) (fine lines) of a 2D environment, (b) the corresponding generalized Voronoi graph (GVG) with vertices placed at the position of the corresponding meet points for visualization.

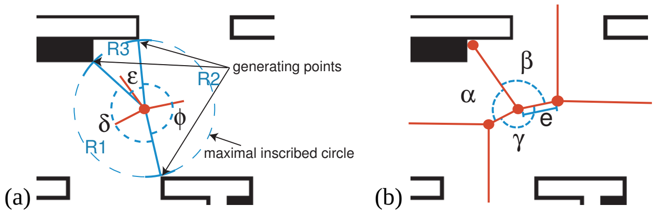

Figure 2. Different kinds of annotations to the GVG-based route graph: (a) Vertex signature containing the distance of the generating points (radius of the maximal inscribed circle) and angles between the connections to the generating points, (b) relative position information given by the angles between departing edges and the length of the edges.

introduced in [2] and has been first used in robotics as an intermediate representations to solve motion planning tasks given complete information about the working space of the robot (usually by providing a geometric description of the boundaries of the obstacles) [13, 8]. In the two-dimensional case it contains all points of free space that are center of maximal inscribed circles (circles that are maximally expanded without intersecting the obstacle boundaries) that touch the obstacle boundaries at at least two points. Figure 1a shows a simple two-dimensional environment and the corresponding GVD (fine lines) consisting of curves that intersect at meet points and end up in corners of the environment.

As described in [3] simple motion behaviors to follow a Voronoi curve from one meet point to the next or get from a point aside the GVD to one on the GVD can be defined. Therefore, the generalized Voronoi graph (GVG), which is the graph corresponding to the GVD (see figure 1b) with vertices corresponding to meet or corner points and edges connecting vertices joined by Voronoi curves, is well-suited to serve as a topological map [3, 17]. The robot can travel from one vertex of the GVG to any other by repeatedly applying the Follow-Voronoi-Curve behavior while keeping track of the robot's position within the graph structure. The only additional information required for this is the clockwise order of departing edges for each vertex. It is also possible to construct this kind of representation autonomously during an exploration by tracing the Voronoi curves with the same behavior and registering the meet points encountered together with their departing edges.

However, in real world applications noisy and discrete sensor data together with the instability of the underlying GVD, which may show additional or missing vertices and edges if the boundary information changes slightly, require to store more information about the environment and more complex procedures to make the approach robust enough to be applicable. In addition, to construct a complete GVG by tracing every single Voronoi curve of the GVD is quite costly and can be avoided by making better use of the sensor data.

To overcome these problems, we developed a representation that extends the GVG with additional annotations to vertices and edges together with procedures for localization and incremental construction for a robot equipped with a laser range finder [15]. We will briefly describe this approach in the following.

2.1 The GVG-based representation

Besides the graph structure and the clockwise order of edges the following information is contained in our representation:

1. Vertices are labeled with a *signature* (see figure 2a) that contains the angles and distance to the generating points (those points where the maximal inscribed circle centered on the Voronoi meet point touches the obstacle boundaries). This information can be used to distinguish vertices.
2. The approximate relative position of vertices is represented by annotating the edges with the approximate distances and vertices with the approximate angles between departing edges (see figure 2b).
3. Every edge is annotated with a description of the Voronoi curve corresponding to this edge since this curve may deviate from the direct connection between the two vertices.
4. Additional information about which edges are traversable (not too close to obstacles) for the robot and which edges lead to still unexplored areas is also annotated to the graph structure.

In the remainder of the text we will for simplicity still call this extended representation a GVG.

2.2 Path planning, localization, and incremental construction

Path planning in this extended GVG can still be done by applying standard graph search techniques which now might employ the annotations, for instance to plan shortest paths. For execution of a planned path and for incrementally constructing this representation during exploration a robust localization within the graph structure is required.

Therefore, we developed a localization scheme that compares a *local* GVG computed from a single $360°$ scan taken from the robot's current position with the (partially constructed) *global* GVG to identify correspondences taking into account the vertex signature, the relative position information, and odometry information about the last movement. The local GVG describes how the GVG looks in the neighborhood of the robot's current position as far as this can be derived from the sensor data available from this point. The comparison scheme searches for a similar subgraph instead of an exact isomorphism. This way robust localization can be achieved despite the instability caused by the imperfect sensors.

Incremental construction of the global representation is achieved by sequentially merging local GVGs computed for different positions starting with the local GVG computed for the start position of the robot. Thus, it makes maximal use of the information available from each observation avoiding unnecessary exploration steps. It uses the results of the comparison scheme employed for localization to identify parts of the local GVG that can be used to complement the global GVG. The idea of merging local GVGs to construct the global representation is illustrated in figure 3 that shows how a global GVG

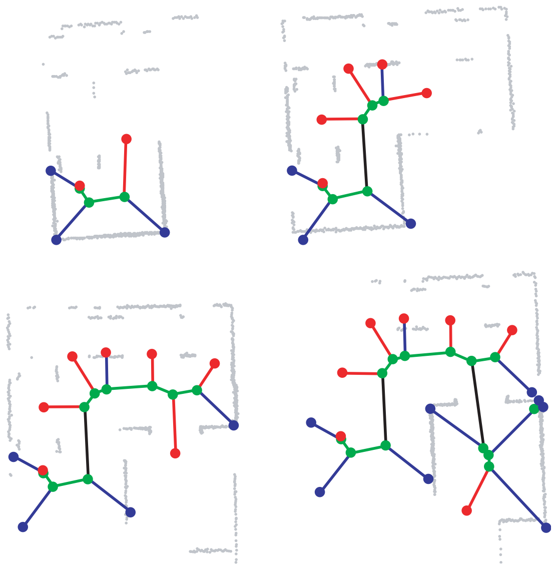

Figure 3. A sequence of growing global GVGs (starting with the local GVG of the robot's start position) constructed during the exploration of an unknown environment.

grows over time (see [15] for details on the localization and construction algorithms).

The GVG-based route graph is a rather compact representation of the essential spatial information required for navigation that can be augmented with additional (e. g. semantic) information if needed. The topological localization approach avoids special efforts required to keep the annotated metric information globally consistent since successful navigation is possible without globally consistent metric information. In addition, the Voronoi-based approach allows systematic exploration of an unknown environment by keeping track of which edges still need to be explored until the GVG is complete. Path planning within the GVG is more efficient than in most metric approaches because the representation only represents qualitative different routes.

3 Hierarchization of the Voronoi-based route graph

Voronoi-based route graphs as described in the previous section contain the information required for successful navigation. However, they also contain details not required for many tasks since not all meet points of the underlying GVD really correspond to decision points relevant for navigation. Some are caused by minor features of the environment like small dents or niches and some are merely the result of noise in the sensor data. For high-level reasoning, planning, and for communication issues a more abstract level of representation would be preferable if it is still linked to the detailed level required for actually acting within the environment. In addition, the relevant vertices are also those that are very stable and thus less likely to be missing in one of the local GVGs. Hence, localization can also benefit from a more abstract level of representation only containing the relevant vertices. Therefore, our goal is to construct a hierarchically structured multi-layer route graph representation that bridges from detailed navigational information to abstract high-level information about the environment and allows to efficiently reason in a hierarchical manner. Every layer of this representation consists of a route graph that models the environment at a certain level of granularity and its features are linked to those of the next higher and next lower layer in a way that allows to switch to a finer or coarser level. To derive more abstract layers from the original GVG a measure is required that assesses the relevance of individual Voronoi vertices for navigation and we will develop such a measure in the next section.

3.1 Assessing the relevance of Voronoi vertices

The GVG as described in section 2 is mainly an undirected Graph RG $= (V, E)$ (with additional annotations) containing only vertices of degree one (the corner vertices) or of degree three or higher (the inner vertices). As figure 2a illustrates, the lines connecting a Voronoi vertex v with its generating points on the obstacle boundaries separate different parts of free space that are accessible via one of v's departing edges. We will call each such area, that can be reached from v without crossing one of the connecting lines again, a region R_i^v of v. For each Voronoi vertex of degree n there exist n such regions. If v is part of a cycle in the route graph, the regions corresponding to the two edges of v that also belong to this cycle will be identical since the edges provide access to the same part of the environment, just from different directions.

How relevant a Voronoi vertex v is for navigation depends directly on its regions. To be regarded as a decision point, at least three of v's regions need to be significant enough to be judged as different continuations after arriving at this point. Otherwise, no real decision is to be made at this point. In addition, having two very significant regions can not make up for the third region being insignificant, e. g. a small niche in a corridor will not create a decision point in front of it irrespective of how long the corridor continues in both direction. Furthermore, having many insignificant regions will not make up for the third most significant one being still insignificant, e. g. two small niches on opposing sides of the corridor will not cause a decision point either.

Therefore, assuming we have a second measure called RSM (for region significance measure) that assesses the significance of each region of v, it makes sense to take the RSM value of v's third most significant region as the relevance value of v. Using maxRSM_3^v to denote the k-highest RSM value of a region of v, we thus define our Voronoi vertex relevance measure (VVRM) for all $v \in V$ with $\text{degree}(v) \geq 3$ as:

$$\text{VVRM}(v) = \text{maxRSM}_3^v.$$

We now need to define the RSM measure in a way that captures the notion of a significant region in the context of navigation in an indoor environment. The two major factors that we wanted to account for in our measure are the following: First, the distance from v to the remotest goals belonging to the region should influence the significance of the region, since a region is clearly more significant if one can reach goals within it that are far from the current position. Second, we wanted to include the aspect of visibility to ensure that a region is assessed as less significant if most of it can be perceived from a larger area around v.

An additional constraint on our measure is that the significance values should be computable from the information contained in the GVG alone without referring to a geometric description of the boundaries of obstacles because this is the only information available in our mapping approach. Furthermore, cyclic regions should be treated as maximally significant so that cycles in the graph will never be split up when deriving a coarser route graph from the GVG (see section 4.1).

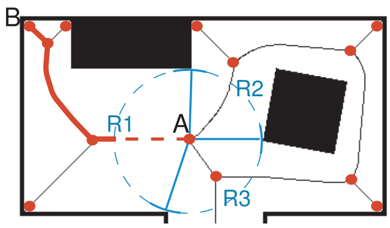

Figure 4. Computation of the RSM value for the region $R1$ to the left of vertex A: The length of the path to B lying within the maximal inscribed circle (dashed line) is subtracted from the length of the complete path to B yielding the length of the solid thick part of the path.

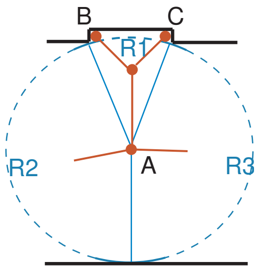

Figure 5. Region $R1$ of vertex A is caused by a small niche in a wall. Since A lies in a large area the distance along the GVD from A to the corner vertices B and C is rather big. However, most part of the connection to these vertices lies within the maximal inscribed circle and will thus be subtracted during the RSM computation resulting in a small RSM value for region $R1$.

Hence, we define the RSM measure as follows:

1. RSM$(R_i^v) = \infty$, if v and the departing edge corresponding to R_i^v belong to a cycle in the route graph.
2. Otherwise, the shortest paths from v to the corner vertices belonging to R_i^v in terms of the distance along the GVD are considered. As illustrated in figure 4, it is determined for which corner vertex the length of this path minus the length of the part of this path lying within the maximal inscribed circle of v is maximal, and this is returned as the value RSM(R_i^v).

In the non-cyclic case the distance of the furthest corner vertex contained in the region is used to measure how far the robot could travel into this region. The subtraction of the length of the part that lies in the maximal inscribed circle of v introduces the notion of visibility as mentioned above. For instance in figure 5 the small niche that causes Voronoi vertex A in a wide hallway will be assessed as insignificant since the most part of the path to B or C is contained within the maximal inscribed circle centered on A meaning most parts of the corresponding region are visible from every point within this circle.

Given the complete global GVG the individual RSM values for a vertex v can be easily computed by applying a slightly modified version of Dijkstra's single source shortest path algorithm [5]. Picking the 3rd highest value then yields VVRM(v). The modifications have to ensure that

- only the edge $e_{R_i^v}$ departing from v into the region $R_i^v$is considered connected to v when relaxing the edges of the start vertex in the first step, and
- cycles leading back to v via a different edge than $e_{R_i^v}$ will be detected and a value of ∞ will be returned.

With the worst-case time complexity of the dominating shortest path algorithm being $O(|E| + |V| \log |V|)$, we end up with a total time complexity for computing VVRM(v) with degree$(v) = n$ of $O(n(|E| + |V| \log |V|))$.

The computation of the relevance values as described above assumes that a complete GVG is available. However, when we want to compute the relevance values for the vertices in a local GVG or for a still only partially constructed global GVG, we have to adapt this approach: We then treat all vertices that mark the boundary of the explored area like corner vertices and compute the relevance values as before. Doing this, all RSM values for regions which contain edges marked as unexplored and which do not correspond to a cycle in the graph will just be lower bounds on the true significance value of the region and will be marked as such. VVRM(v) of any vertex v only yields the exact relevance value if all RSM values for this vertex from the 3rd highest on are exact values and not just lower bounds. Otherwise, the fact that one of these RSM values could actually be higher could result in a higher 3rd highest RSM value for this vertex. Thus, VVRM(v) in this case is also just a lower bound on the real relevance value of v. Lower bound estimates of relevance values are updated when more information allows to make an estimate closer to the true relevance value and such vertices are treated like vertices with an relevance value of ∞ when deriving a coarser route graph layer, as long as the exact value cannot be determined.

3.2 Coarser level route graphs and abstraction relation

Unfortunately we can only give a short description on how the simplification algorithm that derives a coarser route graph from the original route graph works here. The algorithm removes every vertex v with an relevance value that is not higher than a given threshold θ together with the subgraphs that correspond to the regions of v that are classified as insignificant by the RSM value. In certain cases it becomes necessary to replace v by a new vertex of degree one to ensure that every relevant vertex has a departing edge for every significant region accessible from it. Replaced substructures of the original GVG will be represented either by a single vertex or a single edge in the coarser route graph.

Figure 6 shows the result of applying this algorithm to the GVG previously shown in figure 1. θ was set to 1000mm in this example, a value that already produces quite abstract representations since most of the vertices caused by small dents and niches are removed. Figure 7 illustrates how parts of the original GVG (shown at the bottom) are represented by a vertex or an edge in the coarser route graph (top). When we are building up the multi-layer representation corresponding features of adjacent layers are linked as indicated by the arrows. Thus, we have two kinds of edges in our hierarchical representation: route graph edges horizontally connecting vertices within the same route graph layer and abstraction edges vertically connecting vertices and edges in one layer with subsets in the layer below.

4 Path planning, reasoning, and communication with the hierarchical GVG-based route graph

In this section we point out on how we think path planning, spatial reasoning, and communication about spatial information can benefit from the hierarchical route graph representation described in the previous section.

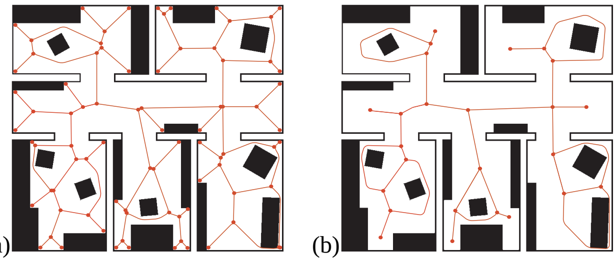

Figure 6. Results of the simplification algorithm: The original GVG (a) is transformed into a coarser route graph (b).

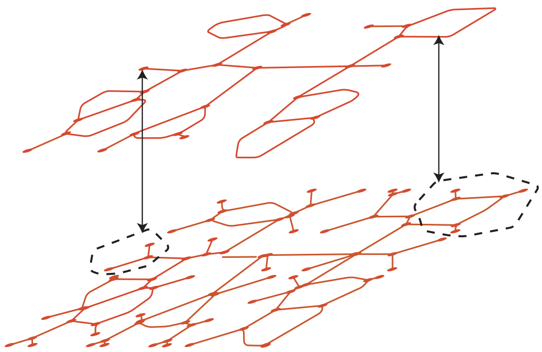

Figure 7. A two-layer hierarchical route graph representation with the original GVG at the bottom and a coarser route graph layer on top of it. Two examples of how parts of the detailed level are represented by a vertex or an edge of the coarser level are shown by the arrows.

4.1 Path planning

A hierarchical route graph representation like the two-layer example from the previous section can be employed for hierarchical path planning. The edges in the coarse layer in a way correspond to macro operations like driving from one door to the next along a corridor or passing an object on one side. Thus, planning on the high-level (e.g. by using graph search techniques) results in a plan that is not directly executable with the low-level navigation procedures of the robot. However, the abstraction relation allows to recursively break down more abstract operations into finer operations until a plan at the detailed level of the original GVG is reached.

The definition of the relevance measures and the simplification algorithm used assure that cycles in the original GVG are either retained at a coarser level or that a complete subgraph containing the cycle is replaced by a vertex or an edge. But a cycle will never split up when changing to a higher level of abstraction. This guarantees that a shortest path planned on a higher level will always result in the shortest path at the bottom level as well, when it is recursively transformed into an executable plan.

4.2 Spatial Reasoning

In [11] we described an approach to reason about the relative positions of the decision points within the low-level GVG-based route graph by propagating intervals for the distance and angles (called distance-orientation intervals (DOIs)) annotated to the graph structure along the sequence of edges connecting two vertices. This approach is similar to the composition of spatial relations in qualitative spatial reasoning [4]. The DOIs represent the uncertainty in the metric relative position information assuming certain maximal error boundaries. Reasoning about relative positions of the decision points in the route graph can for instance be applied to determine potential candidates for loops in the environment that need to be closed while constructing the representation during an exploration. Another application is judging if an unexplored junction in a partially constructed route graph might be a good shortcut to a place visited earlier.

This reasoning about routes can also benefit from the hierarchical organization of the route graph representation. Intermediate results from the low-level propagation can be stored as relative position information at the higher levels. This would allow to employ a hierarchical propagation scheme that uses the distance orientation intervals at the highest level if they are available or switches to a lower level whenever this is not the case, adding the result to the higher level after it has been computed. On the long run the higher level will be completely annotated speeding up the relative position computation significantly.

4.3 Communication

The most abstract route graph layer in our representation provides a compact description of the environment that is rather independent of the particular properties of the range sensor of the robot that constructed the representation. Therefore, this information is much better suited to be communicated to another spatial agent than the detailed description given by the original GVG. Scenarios that come to mind here are multi-robot exploration scenarios in which the individual robots exchange knowledge about parts of the environment they have explored so far. However, there are of course limits to the degree of difference in the sensors that can be dealt with without further developing reasoning mechanism to handle e. g. problems arising from differences in the individual GVGs caused by obstacles that can be seen by one robot but not the other due to different heights in which the range sensors are mounted.

Another application scenario in which the abstract route graph level can be employed beneficially is human-robot communication about routes. Augmenting the route graph with semantic information for instance stemming from door recognition modules will allow the robot to generate route instructions to guide a person to a certain goal. In addition, such a representation will make it easier to match a route description given by a human instructor to the robot's model of the environment and translate it into a detailed sequence of actions, since the abstract route graph with all irrelevant vertices and edges removed will be much closer to the route graph the instructor had in mind when generating the description.

5 Experimental results

In first experiments we tested the relevance measures and the simplification algorithm on real data collected with our Pioneer 2 robot while it was driving along a corridor in our office building. Figure 8a shows a section of the GVG constructed during this exploration run. 8b shows the route graph computed from this GVG (again for a threshold value of 1000mm). It demonstrates how the algorithm successfully removes vertices and edges caused by small dents or noise resulting in a route graph that only contains edges for traveling along the corridor and for entering the rooms on both sides.

In a second experiment we used a simulation to perform two exploration runs with different noise ratios in the range sensor data. Applying the simplification algorithm to both GVGs constructed

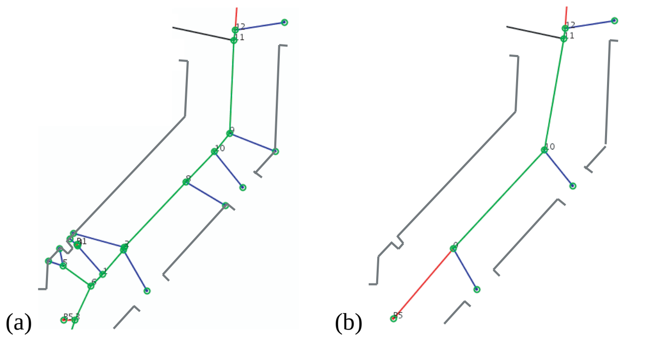

Figure 8. Example of a coarse route graph (b) computed from a GVG constructed with a real robot that drove down a corridor with flanking offices.

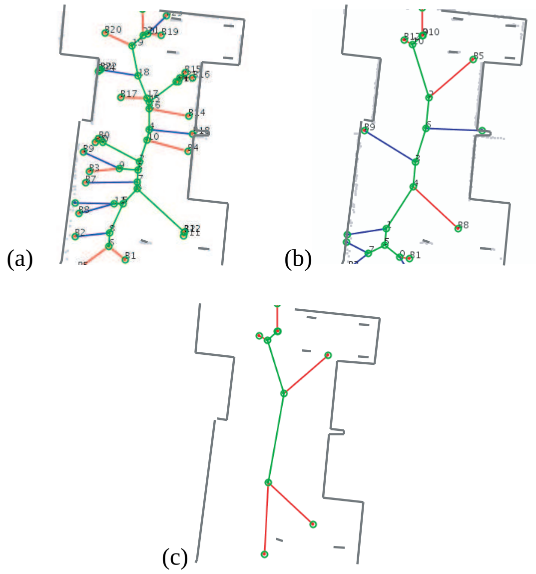

Figure 9. Simulation of different sensor properties: (a) shows the GVG of a robot with high and (b) of one with low sensor noise. Identical coarse route graphs are computed from both GVGs (c).

during those runs (shown in figure 9a and 9b) resulted in identical route graphs for both cases shown in 9c, only varying slightly in the exact positions of the vertices. This demonstrates that our coarser route graph representation is better suited to allow multiple robots equipped with different range sensors to exchange spatial knowledge than the original GVGs.

6 Conclusions

We have proposed a hierarchically organized Voronoi-based route graph representation for robot navigation and exploration tasks in office-like indoor scenarios. The representation bridges the gap between low-level spatial information for navigation and abstract route-based representations well-suited for high-level planning and spatial reasoning. We showed how such a representation can be constructed using a Voronoi vertex relevance measure and how it can be employed for hierarchical planning, spatial reasoning, and robot-robot or human-robot communication. We hope to further explore these applications in the future. In addition, we plan to address other issues involved in generating suitable abstract route graphs like the fact that multiple Voronoi vertices located close to each other may be treated more adequately as a single decision point, something that is important for human-robot communication.

ACKNOWLEDGEMENTS

The author wishes to thank Christian Freksa, Reinhard Moratz, Frank Dylla, and Diedrich Wolter as well as two anonymous reviewers for valuable comments and suggestions. This research was supported by the German Research Foundation (DFG) grant SFB/TR 8 ’Spatial Cognition’.

REFERENCES

[1] F. Aurenhammer, ‘Voronoi diagrams - a survey of a fundamental geometric data structure’, *ACM Computing Surveys*, **23**(3), 345–405, (1991).

[2] H. Blum, ‘A transformation for extracting new descriptors of shape’, in *Models for the Perception of Speech and Visual Form*, ed., W. Wathen-Dunn, pp. 362–380, Cambridge, MA, (1967). M.I.T. Press.

[3] H. Choset and J. Burdick, ‘Sensor based exploration: The hierarchical generalized Voronoi graph’, *The International Journal of Robotics Research*, **19**(2), 96–125, (2000).

[4] A. G. Cohn, ‘Qualitative spatial representation and reasoning techniques’, in *KI97: Advances in Artificial Intelligence. Proceedings of the Twenty-first Annual German Conference on Artificial Intelligence*, eds., G. Brewka, C. Habel, and B. Nebel, pp. 1 – 30. Springer; Berlin, (1997).

[5] E. W. Dijkstra, ‘A note on two problems in connection with graphs’, *Numerische Mathematik*, **1**, 269–271, (1959).

[6] D. G. Kirkpatrick, ‘Efficient computation of continuous skeletons’, in *20th Annual Symposium on Foundations of Computer Science*, pp. 18–27. IEEE, (1979).

[7] B. J. Kuipers and Y.-T. Byun, ‘A robust, qualitative method for robot spatial learning’, in *AAAI 88. Seventh National Conference on Artificial Intelligence*, pp. 774–779, (1988).

[8] J.-C. Latombe, *Robot Motion Planning*, Kluwer Academic Publishers, 1991.

[9] D. T. Lee and Robert L. Drysdale III, ‘Generalization of Voronoi diagrams in the plane’, *SIAM Journal on Computing*, **10**(1), 269–271, (1981).

[10] F. Lu and E. Milios, ‘Robot pose estimation in unknown environments by matching 2d scans’, in *IEEE Computer Vision and Pattern Recognition Conference (CVPR)*, (1994).

[11] R. Moratz and J. O. Wallgrün, ‘Spatial reasoning about relative orientation and distance for robot exploration’, in *Spatial Information Theory: Foundations of Geographic Information Science. Conference on Spatial Information Theory (COSIT)*, eds., W. Kuhn, M.F. Worboys, and S. Timpf, Lecture Notes in Computer Science, pp. 61–74. Springer Heidelberg, (2003).

[12] H. Moravec and A. Elfes, ‘High resolution maps from angle sonar’, in *Proceedings of the IEEE Conference on Robotics and Automation*, pp. 116–121, (1985).

[13] C. Ó’Dúnlaing and C. K. Yap, ‘A retraction method for planning the motion of a disc’, *Journal of Algorithms*, **6**, 104–111, (1982).

[14] S. Thrun, D. Fox, and W. Burgard, ‘A probabilistic approach to concurrent mapping and localization for mobile robots’, *Machine Learning*, **31**, 29–53, (1998). also appeared in Autonomous Robots 5, 253–271 (joint issue).

[15] J. O. Wallgrün, ‘Exploration und Pfadplanung für mobile Roboter basierend auf Generalisierten Voronoi-Graphen’, Diploma thesis, Fachbereich Informatik, Universität Hamburg, (2002).

[16] S. Werner, B. Krieg-Brückner, and T. Herrmann, ‘Modelling navigational knowledge by route graphs’, in *Spatial Cognition II*, volume 1849 of *LNCS,LNAI*, 295–316, Springer, (2000).

[17] D. Van Zwynsvoorde, T. Simeon, and R. Alami, ‘Building topological models for navigation in large scale environments’, in *2001 IEEE International Conference on Robotics and Automation (ICRA’2001)*, pp. 4256–4261, (2001).