



Development of a Shared Environment Model with Dynamic Trajectory Representation for Multiple Mobile Robots

Sebastian Q. Roder¹, Florian Rothmeyer^{1,2}, Bernd Spiegelberger², and Gunther Reinhart¹

¹ Institute for Machine Tools and Industrial Management,
Technical University of Munich, Munich, Germany

Sebastian.Roder@iwb.tum.de

² ITQ GmbH, Parkring 4, DE-85748 Garching, Germany

Abstract. The productivity of groups can be increased by enabling group members to share their perceptions of the environment. We adapt this concept for mobile robots by presenting an object-oriented approach to a shared environmental model. The objects are stored in a graph, which saves memory and computing power and allows the representation of hierarchical and topological relationships. Each object can contain geometric and semantic data as well as information about its current, past, and planned or estimated future movements. An example application shows that modeling future motion can prevent collisions.

Keywords: world models, distributed systems, dynamic objects, collaborative robots

1 Motivation

Product customization and shortening of production life cycles increase the demand for flexible manufacturing systems. Mobile robots show great potential due to their high configurability and scalability as well as their freedom of movement. The applications are diverse and range from logistics and cleaning or inspecting to operation at multiple workstations. The large number of mobile resources means that future production facilities will require decentralized intelligence to handle the increasing complexity.

In order to navigate and interact safely and efficiently in such facilities, robots need an understanding of their environment that is technically enabled by environmental models. The combination of the limited perceptions of individual robots into a common knowledge base enables collective intelligence. Those shared models enable improvements firstly in safety, e.g. warning of imminent danger, and secondly in efficiency, e.g. avoiding traffic jams.

Existing models lack the ability to represent object dynamics. At critical points in a production facility, such as intersections and pedestrian crossings, information about the dynamics of surrounding objects can improve overall performance and safety. Dynamic information could either be directly integrated

into the map by the performing robot or estimated by observers for agents with no connection to the shared model, e.g. human workers.

This work presents a novel approach for integrating object dynamics into world models. The suggested solution consists of a client-server architecture and a graph-based data model supporting geometry, semantic classification, topology and object dynamics, including past and present as well as predicted and planned future trajectories. First, an overview of environment models with focus on papers related to this work is given in section 2. Then, our approach is presented in section 3. Section 4 describes a proof-of-concept application offering a technical implementation and illustrating the added value of the approach. Finally, the results are discussed in section 5 and perspectives for future research are given.

2 Related Work

Research on shared environmental models focuses on three main areas: methods for obtaining new data and integrating them into an environmental model, solutions for contributing and distributing data in multi-user scenarios and lastly, the actual data modeling, which uses data structures, itself.

Obtaining New Data This area includes the various methods from the fields of perception, data fusion and pattern recognition and is considered only briefly below. Standard approaches to mapping in mobile robotics are “Simultaneous Localization and Mapping” (SLAM) [1, 2] or “HectorSLAM” [3], which generate grid-based maps. Alternatively, the raw sensor data can also be processed by object recognition and integrated into an object-oriented data model [4]. There are several applications in the field of service robotics [5, 6]. Furthermore, data can be processed to form topological graphs, e.g. transforming a 2D grid-based map of a building into a graph with nodes representing rooms [5, 7]. Applications to path planning for multiple mobile robots are given in [8]. Due to inaccuracy in perception, objects need to be registered to their absolute position in the environmental model, referred to as “anchoring” or “data association” [9, 10]. Context-based classification improves anchoring by considering existing knowledge [11].

Data Distribution This area integrates the methods from the fields of network architecture, consistency management and synchronization and, again, is considered only briefly below. In contrast to fully centralized map storage and fully synchronized maps with all participants, intermediate stages are a trade-off between latency and consistency [12]. A client-server structure can be applied to an object-oriented distribution model, preserving consistency by predefined access rights to object changes [13]. Multi-server architectures with environmental subsets enable scalability [14]. Moreover, there are approaches that are tolerant to inconsistency by assuming it to be acceptable at certain points of the data model [15].

Data Structures for Environment Mapping Our approach focuses on data structures, therefore the related work is discussed in detail. Common approaches for geometric relationships can be classified as grid-based or object-oriented. In addition to geometric data, environmental models can represent semantics or uncertainty.

Grid-based Structures A fundamental concept of map generation is the “Occupancy Grid Map” (OGM), discretizing 2D or 3D space into independent cells assigned with the estimated occupancy, e.g. “free”, “occupied” or “unknown” [1, 3, 5, 16]. The storage effort can be reduced by structuring the grid cells in a hierarchical manner, the so-called “octree” [17]. Combinations of several octrees enable object hierarchies and multiple resolutions [18].

Object-oriented Structures With object trees, not only cubes, but any geometric object can be described. Gregor *et al.* [19] use homogeneous transformations to indicate the relative spatial arrangement of parent and child objects. Semantic information can be represented by linking the nodes of the object tree to corresponding elements of a “conceptual hierarchy” [5].

An extension to graph structures holding geometric and transformation nodes is presented by Blumenthal *et al.* [20]. Here, nodes can contain semantic information via key-value pairs and past object movements are traced by transformation-timestamp pairs. The graph structure allows several transformation nodes to point to the same geometry node, which enables the modeling of competing perceptions in multi-robot scenarios. Multi-user operation is considered unproblematic because geometric objects, as well as transformation-timestamp pairs, are invariant. In addition, there is an interface for distributed storage of a graph, but possible consistency problems are not considered. Other approaches extend geometric models by separately stored ontologies [10, 21], which are not in the scope of this work.

Further graph structures, such as in [22], focus on the shared use of the map by several actors. For this purpose, a so-called “generative grammar” is set up, which regulates how parts of the model may change. The emphasis of the research can also be on a systematic class-based representation of geometric information [23].

Modelling of Uncertainty Predictions or measurements result in uncertain data, which must be correctly modeled. By augmenting coordinates with one standard deviation per dimension, locations can be stored as Gaussian distributions [24]. Likewise, uncertain occupancies in octrees can be expressed by probability values [17] and spatial transformations can be extended by additional covariance matrices, expressing 6D spatial deviation [20].

Different approaches introduce confidence values for graph attributes and links [10]. Object classification algorithms provide a classification probability, which is integrated into the class structure by Papp *et al.* [23]. Additionally, generic uncertainty models can be linked to physical quantities here.

3 Scientific Approach

The overview of the scientific approaches to environmental modeling shows that so far there are only solutions that are specialized in one field. If several robots with a shared environmental model are used, their activities and perceptions are very homogeneous. This in turn leads to restrictions in the representation of the perceived world in the model. Comprehensive environmental models, on the other hand, are not prepared for multi-user or distributed systems. It is not possible to enter predicted trajectories of robots or dynamic obstacles into the world model in any of the considered solutions.

3.1 Requirements

From the shortcomings of the existing solutions mentioned in the previous section and the generally necessary characteristics of an environmental model, we can derive some requirements for a new, comprehensive system for mapping the environment.

Geometry and Dynamics Real world objects should be depicted in terms of their position and shape. In order to adequately reproduce movements of objects, a distinction must be made between past, present and future movements. The latter must be further subdivided into planned and estimated movements.

Combination of Topological, Geometrical and Hierarchical Relations Three types of relations are defined in the literature:

- hierarchical, indicating a dependency of the function and location of subordinate objects on their parent,
- geometrical, indicating a spatial relation, and
- topological, indicating neighborhoods that are not necessarily directly connected to the spatial location.

For compact storage, a single data structure should represent these three types of relations.

3.2 Scientific Concept

Our system is based on a client-server architecture similar to those in [13] and [14]. Each instance contains a copy of the shared environmental data. In the following, we describe how the data is structured and represented, focusing on motion modeling.

Structure of the Data Model Our approach is based on object-oriented representation. Geometric or abstract objects of the real world are mapped to nodes of a data structure whose edges reflect relationships between the objects.

The data structure has two types of object links, vertical and horizontal. The relative position of two objects is stored in vertical links. They form a tree-like structure and therefore also show hierarchies between objects. In contrast to a conventional tree, objects can have more than one parent node in the superordinate level, resulting in a directed acyclic graph (DAG). This means that multiple memberships can be mapped, for example a workpiece that is jointly transported by two robots. If an object has more than one parent node, its absolute position is ambiguous because there are several paths to the root node.

The second type of object connection is the horizontal link. It shows the topological neighborhood of two objects. Horizontal links may interconnect any objects in the DAG of vertical links. The collection of all horizontal links in a data model forms one or more graphs.

Contents of the Data Model Each node of the data model represents a real physical object or a group of several subordinate objects [18, 20]. Thus a node does not necessarily have to contain a geometry. The assignment of a classification and a trajectory is also possible so that all use cases can be covered.

Geometry and Classification In our approach, several types of geometry representation can be stored. Point clouds provide a storage option close to the sensor, while octrees provide a storage-efficient solution. Moreover, triangle meshes can be stored, which are very suitable for the visualization of objects. The technical details are not discussed in detail here; instead, readers are referred to the literature [17, 20].

In addition to geometries, the results of classification algorithms can be stored in the nodes [10, 23].

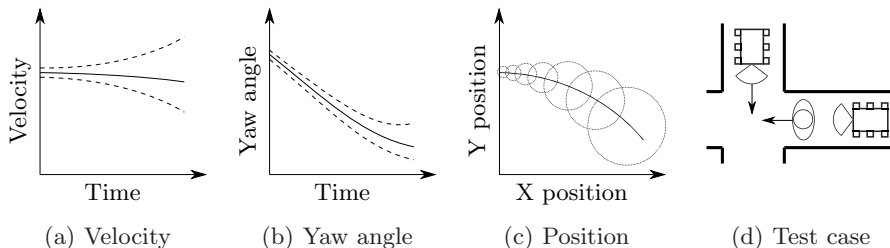


Fig. 1. (a, b) Dispersing trajectories of velocity and orientation (yaw) angle plotted as mean value and standard deviation (dashed) [25] (c) Discrete trajectory of uncertain positions [25] (d) Test-setup of first robot (top), human worker (middle) and observing robot (right)

Object Dynamics Present and past object dynamics are modeled according to [20]. This is applicable to the linear and angular velocity stored in each object node, as well as to the location of objects stored in the vertical links.

For interfering objects, future motions can be predicted based on present and past motions. Our approach includes an uncertainty model for representing predicted trajectories [25]. Figures 1(a) and 1(b) show dispersing trajectories of velocity and orientation angle, expressed as the mean value and standard deviation. Accordingly, discrete trajectories are formed by time-sampled points where the position is assigned an uncertainty in the form of a covariance matrix, see Figure 1(c). This representation of an uncertain trajectory has the advantage that time and location components are separated. In an environmental model which is based on geometric relationships, the trajectory can therefore be seamlessly inserted.

In case of robots, future motions are predictable. Therefore, our approach includes planned trajectories as continuous paths (NURBS: Non-Uniform Rational B-Spline) mapped to time.

4 Proof of Concept

A typical system has been created using open-source software. Due to the early stage of development, an industry-like application has been simulated in an office environment, e.g. an order picking robot in a small parts warehouse.

4.1 Implementation

The described object-oriented data model has been written in C++. Several hierarchical classes represent the different nodes and edges as well as the optional information about classification, geometry and trajectory. They are depicted on the left side of Figure 2. Linux-based client programs were implemented for reading sample geometries from a database, inserting a trajectory and visualizing the model. Another ROS-based client makes the world model accessible to the mobile robot.

The clients communicate via a WLAN with a linux-based server application. If a client makes a change to its instance of the data model, it informs the server. The server forwards the change to the other clients.

4.2 Experiment and Observation

The use example consists of a mobile robot driving along a narrow alley. At a hard-to-observe intersection a human worker crosses the robot's pathway.

The robot, which is only equipped with on-board sensors, would need to reduce its speed at such locations to ensure safety. Therefore the robot's performance decreases even if no interference occurs.

In this scenario it is assumed that a second robot beyond the junction can perceive the person and estimate their future movement. The expected trajectory

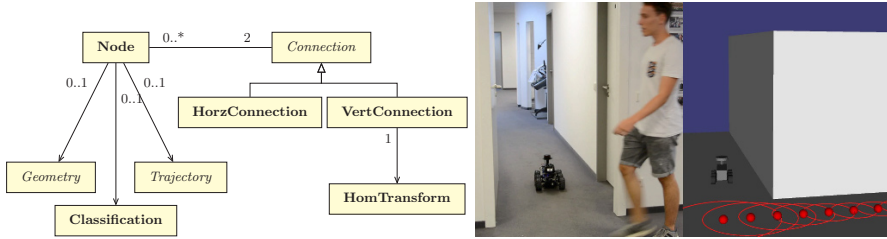


Fig. 2. Classes of the implemented data model (left), typical view of the experiment (middle) and matching environment model including predicted trajectory as circles on the floor (right)

is fed into the environmental model available to the first robot. The setup is illustrated in Figure 1(d).

Experiments prove that the first robot can react to the interference by stopping before being able to perceive the human worker based on onboard sensors alone. Figure 2 shows the crossing situation in reality and in the model.

5 Results, Discussion and Conclusion

In this work, a novel shared environmental model for mobile robots was presented. The graph-based data model can represent geometry as point clouds or triangle meshes and semantic information as classification results. The proposed solution is suitable to represent future or past object dynamics and to consider the uncertainty in predicted trajectories, as demonstrated in the section above.

Furthermore, the environmental model offers advantages in the operation of mobile robots. By communicating with other robots, a robot can react to a moving obstacle before it has detected it with its on-board sensors.

Robots must normally drive at an appropriate speed to be able to brake at dangerous points such as hard-to-observe intersections. It is also conceivable to install stationary sensors at danger points so that potentially interfering objects are always reported to the shared environmental model. Thus overall performance can be increased while maintaining safety.

5.1 Fulfillment of Requirements

Current movements can be stored in the model as single values with a timestamp. When a value with a newer timestamp is added, the previous one moves to the history. Future movements can be stored continuously or discretely with uncertainty. In this way, the degree of detail and the time horizon can be varied arbitrarily. This covers a large number of applications.

Combining hierarchies and location relationships makes the modeling of location-independent hierarchies more difficult. On the other hand, it simplifies changing the location of object groups as only the position of the parent node must be updated.

5.2 Outlook

Further Aspects of Modeling for Multiple Robots In addition to the aspects described above, the presented concept contains mechanisms to ensure consistency and efficient distribution of data. These are of great importance in a shared environmental model. The mechanisms are based on the existence of a central, superordinate server that decides on the approval and distribution of updates. To limit network traffic, it is possible to subdivide the environmental model into several regions. In this way, each robot receives updates matched to its location instead of receiving updates for the whole map. The horizontal links between the model objects that are described above are decisive for the subdivision.

Future Work Several possibilities for geometry storage were mentioned in sec. 3.2. It should be investigated whether further methods need to be added to this selection. Since most types of information occurring in a virtual environment are uncertain, the modeling of uncertainty should also be further investigated. Finally, tests of the environmental model with large amounts of content must be carried out to assess storage efficiency and speed so that the results can be compared with other solutions.

6 Acknowledgements

The results presented in this article were developed within the FORobotics (AZ-1225-16) research network. The authors would like to thank the Bavarian Research Foundation and all participating project partners for their funding and support of the project.

References

1. Chatila, R. & Laumond, J. *Position referencing and consistent world modeling for mobile robots in 1985 IEEE International Conference on Robotics and Automation Proceedings* 1985 IEEE International Conference on Robotics and Automation Proceedings. **2** (Mar. 1985), 138–145.
2. Durrant-Whyte, H. & Bailey, T. Simultaneous localization and mapping: part I. *IEEE Robotics Automation Magazine* **13**, 99–110 (June 2006).
3. Kohlbrecher, S., Stryk, O. v., Meyer, J. & Klingauf, U. *A flexible and scalable SLAM system with full 3D motion estimation in 2011 IEEE International Symposium on Safety, Security, and Rescue Robotics* 2011 IEEE International Symposium on Safety, Security, and Rescue Robotics (Nov. 2011), 155–160.
4. Mason, J. & Marthi, B. *An object-based semantic world model for long-term change detection and semantic querying in 2012 IEEE/RSJ International Conference on Intelligent Robots and Systems* 2012 IEEE/RSJ International Conference on Intelligent Robots and Systems (Oct. 2012), 3851–3858.

5. Galindo, C. *et al.* *Multi-hierarchical semantic maps for mobile robotics in 2005 IEEE/RSJ International Conference on Intelligent Robots and Systems* 2005 IEEE/RSJ International Conference on Intelligent Robots and Systems (Aug. 2005), 2278–2283.
6. Marton, Z.-C., Pangercic, D., Blodow, N. & Beetz, M. Combined 2D–3D categorization and classification for multimodal perception systems. *The International Journal of Robotics Research* **30**, 1378–1402 (Sept. 1, 2011).
7. Goerke, N. & Braun, S. *Building semantic annotated maps by mobile robots in Proceedings of the conference towards autonomous robotic systems* (2009), 149–156.
8. Binder, B. *Spatio-temporal prioritized planning* Diplomarbeit (Technische Universität Wien, 2017).
9. Coradeschi, S. & Saffiotti, A. *Anchoring symbols to sensor data: preliminary report in Proceedings of the Seventeenth National Conference on Artificial Intelligence and Twelfth Conference on Innovative Applications of Artificial Intelligence* (2000), 129–135.
10. Baum, M., Gheȃa, I., Belkin, A., Beyerer, J. & Hanebeck, U. D. *Data association in a world model for autonomous systems in 2010 IEEE Conference on Multisensor Fusion and Integration* 2010 IEEE Conference on Multisensor Fusion and Integration (Sept. 2010), 187–192.
11. Günther, M., Ruiz-Sarmiento, J. R., Galindo, C., González-Jiménez, J. & Hertzberg, J. Context-aware 3D object anchoring for mobile robots. *Robotics and Autonomous Systems* **110**, 12–32. ISSN: 0921-8890 (Dec. 1, 2018).
12. Fleury, C., Duval, T., Gouranton, V. & Arnaldi, B. *A new adaptive data distribution model for consistency maintenance in collaborative virtual environments* in. Joint Virtual Reality Conf. of EGVE 2010 - The 16th Eurographics Symposium on Virtual Environments, EuroVR 2010 - The 7th EuroVR (INTUITION) Conf., VEC 2010 - The Annual Virtual Efficiency Congress (2010), 29–36.
13. Singh, G., Serra, L., Png, W., Wong, A. & Ng, H. *BrickNet: sharing object behaviors on the Net in Proceedings Virtual Reality Annual International Symposium '95* Proceedings Virtual Reality Annual International Symposium '95 (Mar. 1995), 19–25.
14. Hesina, G., Schmalstieg, D., Fuhmann, A. & Purgathofer, W. *Distributed open inventor: A practical approach to distributed 3D graphics* in *Proceedings of the ACM symposium on Virtual reality software and technology* (1999), 74–81.
15. Kharitonov, V. Y. *A Consistency Model for Distributed Virtual Reality Systems in 2009 Fourth International Conference on Dependability of Computer Systems* 2009 Fourth International Conference on Dependability of Computer Systems (June 2009), 271–278.
16. Shaik, N., Liebig, T., Kirsch, C. & Müller, H. in *KI 2017: Advances in Artificial Intelligence* (eds Kern-Isberner, G., Fürnkranz, J. & Thimm, M.)

- Lecture Notes in Computer Science* 10505, 249–261 (Springer International Publishing, Cham, 2017). ISBN: 978-3-319-67189-5.
17. Wurm, K. M., Hornung, A., Bennewitz, M., Stachniss, C. & Burgard, W. *OctoMap: A probabilistic, flexible, and compact 3D map representation for robotic systems* in *Proc. of the ICRA 2010 workshop on best practice in 3D perception and modeling for mobile manipulation* **2** (2010).
 18. Wurm, K. M. *et al.* *Hierarchies of octrees for efficient 3D mapping in 2011 IEEE/RSJ International Conference on Intelligent Robots and Systems* 2011 IEEE/RSJ International Conference on Intelligent Robots and Systems (Sept. 2011), 4249–4255.
 19. Gregor, R., Lutzeler, M., Pellkofer, M., Siedersberger, K.-H. & Dickmanns, E. D. EMS-Vision: a perceptual system for autonomous vehicles. *IEEE Transactions on Intelligent Transportation Systems* **3**, 48–59 (Mar. 2002).
 20. Blumenthal, S., Bruyninckx, H., Nowak, W. & Prassler, E. *A scene graph based shared 3D world model for robotic applications* in. *Proceedings - IEEE International Conference on Robotics and Automation* (2013), 453–460.
 21. Ulbrich, S., Nothdurft, T., Maurer, M. & Hecker, P. *Graph-based context representation, environment modeling and information aggregation for automated driving* in *2014 IEEE Intelligent Vehicles Symposium Proceedings* 2014 IEEE Intelligent Vehicles Symposium Proceedings (June 2014), 541–547.
 22. Manso, L. J. *et al.* *Deep Representations for Collaborative Robotics in Brain-Inspired Computing* (eds Amunts, K., Grandinetti, L., Lippert, T. & Petkov, N.) (Springer International Publishing, 2016), 179–193. ISBN: 978-3-319-50862-7.
 23. Papp, Z., Brown, C. & Bartels, C. *World modeling for cooperative intelligent vehicles in 2008 IEEE Intelligent Vehicles Symposium* 2008 IEEE Intelligent Vehicles Symposium (June 2008), 1050–1055.
 24. Roth, M., Vail, D. & Veloso, M. *A real-time world model for multi-robot teams with high-latency communication* in *Proceedings 2003 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2003) (Cat. No.03CH37453)* Proceedings 2003 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2003) (Cat. No.03CH37453). **3** (Oct. 2003), 2494–2499.
 25. Wiest, J., Höffken, M., Kreßel, U. & Dietmayer, K. *Probabilistic trajectory prediction with Gaussian mixture models in 2012 IEEE Intelligent Vehicles Symposium* 2012 IEEE Intelligent Vehicles Symposium (June 2012), 141–146.

Open Access This chapter is licensed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license and indicate if changes were made.

The images or other third party material in this chapter are included in the chapter's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the chapter's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.

