The 2010 IEEE/RSJ International Conference on
Intelligent Robots and Systems
October 18-22, 2010, Taipei, Taiwan

Dynamic Path Planning Adopting Human Navigation Strategies for a Domestic Mobile Robot

Fang Yuan*, Lukas Twardon* , Marc Hanheide**

***Abstract*— Mobile robots that are employed in people's homes need to safely navigate their environment. And natural human-inhabited environments still pose significant challenges for robots despite the impressive progress that has been achieved in the field of path planning and obstacle avoidance. These challenges mostly arise from the fact that (i) the perceptual abilities of a robot are limited, thus sometimes impeding its ability to *see* relevant obstacles (e.g. transparent objects), and (ii) the environment is highly dynamic being populated by humans. In this contribution we are making a case for an integrated solution to these challenges that builds upon the analysis and use of implicit human knowledge in path planning and a cascade of replanning approaches. We combine state of the art path planning and obstacle avoidance algorithms with the knowledge about how humans navigate in their very own environment. The approach results in a more robust and predictable navigation ability for domestic robots as is demonstrated in a number of experimental runs.**

I. Introduction

In general path planning answers the question *"How can I navigate to a goal position from my current location?"*. Most of the established approaches assume that apriori knowledge about the new environment is available (usually a 2D map representation denoting obstacles and free spaces) and try to compute a plan towards the robot goal [1]. These approaches are purely map-based, without considering any prior experience about navigating the environment. Generally, two types of planning algorithms can be distinguished. The first is to compute a path from the robot's start location to the goal deterministically [2] applying a navigation function, such as NF1 presented in [3]. In contrast, the alternative approaches tackle the problem probabilistically, trying to plan a sequence of actions (a path) associated with all possible positions of the environment to the goal [4]. They consider the probabilistic nature of robot motion and perception [5], [6] explicitly in their planning domain. As discrepancies resulting from dynamic obstacles, such as doors and humans, usually exist between the apriori map and the real environment, local strategies handling those objects are required for a robot to execute the computed path plan. Several classical approaches have been proposed to continuously react to obstacles perceived with robot sensors, e.g. DWA (Dynamic Window Approach) [7] applying constrained search in velocity space to obtain optimal actuator commands, VFH (Vector Field Histogram) [8] employing a two-stage data-reduction process to represent the world model with a polar histogram and select the most suitable steering direction among this histogram with a low polar obstacle density, ND (Nearness Diagram) [9] describing environments (robot and goal location, obstacles, as well as free space) with sectors centered on the robot location and designing robot actions according to predefined situations, and so on. By these means, a robot is able to efficiently fulfill the long-term plan, while reacting to unexpected obstacles quickly.

However the main drawbacks of path planning approaches mentioned above lie in the representation of the environment created with the limited perception from robot sensors and the computational complexity especially for calculating a universal plan in real time.

In this paper, we propose to not only rely path planning on the static configuration of the environment but instead employ the knowledge gained from an analysis of human pathways. Humans sharing an environment with a domestic robot are taken into account with the assumption that path planning might benefit from human strategies. By means of this we seek to overcome the limitation of robot perception. It shall enable the robot to avoid static or potential dynamic obstacles like walls on the hallway or hazards near the doorway for safely planning, and enable it to reach goals more efficiently. Consequently, people are invited in an interactive scenario, the so-called *Home-Tour* (see section III), in which the robot cannot only explore the environment and build a map effectively, but also observe the behavior of the guide-person during the navigation of the environment to positively affect its own navigation abilities. A combination of human-aware path planning, dynamic replanning abilities, and dynamic obstacle avoidance is proposed in this contribution.

The remainder of this paper is organized as follows. After discussing the most recent works on path planning within interactive scenarios in section II, the robot and the designed *Home-Tour* are introduced in section III. Subsequently, details of path planning and the replanning scheme are introduced in section IV. Experimental results are presented in section V, before concluding our work in section VI.

II. Related Work

In 2007, Gockley et al. [10] were one of the the first to propose that knowledge about paths of a human guide could be exploited by robots that needed to make regular trips between specific locations in hospital environments. Path planning for a mobile robot in domestic environment is widely studied, e.g. by Zender et al. [11] and Topp [12]. In

*Applied Computer Science, Faculty of Technology, Bielefeld University, 33615 Bielefeld, Germany, Email: {fyuan, ltwardon}@techfak.uni-bielefeld.de

**School of Computer Science, University of Birmingham, Birmingham, B15 2TT, England, Email: m.hanheide@cs.bham.ac.uk

Fig. 1. The robot **BIRON II** with its hardware components shown on the right. From top right: Pan-/Tilt-camera, interfacial microphone, Pioneer 5DOF arm, and laser range finder.

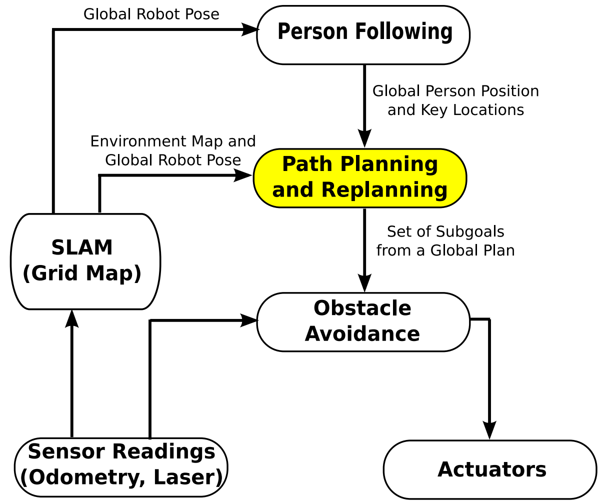


Fig. 2. Overview of the data flow for path planning and replanning. Person trajectory during an interactive scenario and environment information are both observed by the path planning module to compute or correct a plan.

both works a robot followed a person during a guided tour and the geometric features for environment representation were lines extracted from straight structures like walls, without details for small objects. Therefore the built map is not necessarily sufficient for navigation actions. The so-called *Navigation Nodes* leading the robot to goals were selected from the robot trajectory and kept sparse. The robot's trajectory planned through these coarse navigation nodes will be based on the planning in the *static* graph, finding the shortest route. Their representation does not contain any information about the actual routes people walked along. It only creates new nodes if no node already exists within a distance of approximately $1m$. Moreover the choice of sparse nodes for path planning might neglect details from person movement, namely the strategy of the guide-person which could be helpful for obstacle avoidance, such as going through a door. The robot may get into trouble as well, when it is far from those *Navigation Nodes* and without connection to them, or the connections between those nodes are cut off by dynamic obstacles. Therefore, an appropriate approximation and representation of person trajectories, a occupancy-based path planning and replanning abilities are key for robust navigation in these environments.

III. Robot Setup and Scenario

Our approach is integrated with **BIRON II** (see Fig. 1) based on the research platform *GuiaBot*^{TM} by MobileRobots[1]. **BIRON II** is a consequent advancement of the **BIRON** (**BI**elefeld **R**obot compani**ON**) platform [13], which has been under continuous development since eight years. Inside the base there is a laser range finder mounted at a height of about $30cm$ for the perception of the surroundings in front of the robot. Measurements are taken in a horizontal plane covering a $180°$ field of view. The color video camera is mounted for visual perception of the scene and for detailed focusing on persons, areas, and objects. Sound direction can be located with the two interfacial microphones equipped on the top of the robot's body. The upper part of the robot's body houses a touch screen as well as the system speaker used for speech output in user interaction.

As mentioned in section I, we have designed an interactive scenario, the so-called *Home-Tour*. In this scenario a robot is shown around by a user in a real-world apartment and expected to exploit the acquired knowledge, e.g. about the spatial layout and navigation strategies of the guide-person, to autonomously provide services to the users later on. The guide line of the *Home-Tour* is not necessary to be specified. The user can choose an arbitrary route in terms of her or his own decision and spatial understanding of the environment. However, the guide-person should show the places which the robot may visit and are allowed to enter, so that the robot can obtain sufficient information of the new surroundings. Furthermore, the guided tour should be expected to be performed efficiently considering that the user is familiar with her or his own apartment.

As a capability bridging the gap between the movement of the person and the robot especially for the *Home-Tour* scenario we have designed person-following behavior [14]. To realize a following process person tracking is an important prerequisite. The user who is interacting with the robot is tracked with a multi-modal person detection and tracking system from our previous work [15] combining the camera (face detection), laser (legs detection) and microphone data (sound location), and providing person positions in polar coordinate system with respect to the robot. During the *Home-Tour* the human movement containing navigation

[1]www.mobilerobots.com

strategies of the guide-person is observed by the robot. All of the person positions obtaining from the tracking system are recorded to build a graph with multiple layers, as will be elaborated on in section IV-A.

For local navigation the ND (Nearness Diagram) approach tackling cluttered obstacles typically in indoor environments has been integrated into **BIRON II**. As the person positions are defined in a local frame, both the robot poses and the person positions have to be calculated in a common world frame. When the robot knows where it is with respect to the environment, it can navigate and provide services to people. Therefore, the ability of pose estimation in an incrementally built world frame, namely the global representation of the environment makes a mobile robot truly autonomous [16]. A Rao-Blackwellized particle filters [17] based SLAM[2] approach [18] from the library MRPT [19] has been equipped on our robot platform. During a *Home-Tour* the surroundings would be finely represented with grids whose values imply the probability of the occupancy covered by obstacles. As will be presented in section V, the robot will benefit from the occupancy grids not only for the map-based path planning NF1 [20] which calculates a list of free grid cells leading from the robot position to the goal, but also for the perception of dynamic changes, such as a post-closed door of the environment in a replanning process.

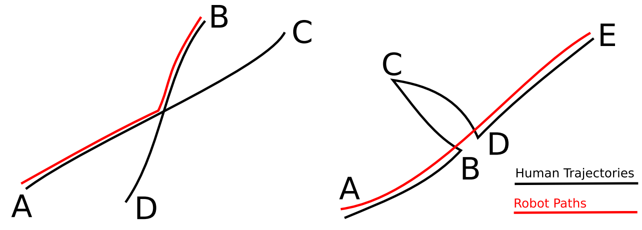
Fig. 2 presents the current system, where the path planning and replanning component marked with yellow provides a sequence of actions, namely the set of subgoals, to the component of obstacle avoidance (ND), in terms of the basic information from the surroundings (the grid map) and the observation of the person behavior during the *Home-Tour*.

IV. Path Planning

Considering the disadvantages of pure map-based approaches as well as static trajectory-based systems discussed in section I and II, we have designed a graph-based planning/replanning approach using the information about the guide-person's behavior. Thus, the robot is expected to benefit from human navigation strategies, as described in section I. Since the search space of our approach is constrained to paths generating from positions of the guide-person during the *Home-Tour*, a map-based path planner has been integrated as a supplement, suppose no plan can be calculated by the graph-based planner.

A. Graph Creation

As discussed in section III, human behavior is observed by the robot during the whole guided tour. The person tracking system provides dense information about person positions calculated in the world frame, namely the occupancy grid map of the environment created by the SLAM component. The naive path planning approach is to let the robot follow the person positions which have been stored during the guided tour in chronological order. Then a path from A to B can be found, only if the user has gone directly from A to B before. Thus, trajectory changes (see Fig. 3(a)) are not possible. Besides, human paths containing unnecessary detours (see Fig. 3(b)) can not be detected.



(a) (b)

Fig. 3. Human Trajectories (black) and Robot Paths (red): In a) the original human trajectories are from A to C and from B to D. If the robot intends to navigate from A to B, a trajectory switch is necessary. In b) the person went from A, via B, C and D to E. If the robot navigates from A to E, the robot does not need to take the detour over C (unless there is an obstacle between B and D, which can be avoided using the path via C).

Hence, the person positions are inserted as nodes N_i into a graph G_{NoEdge} without connections between the nodes. The creation of edges in the graph will be discussed later on in this section. Considering that during the *Home-Tour* introduced in section III different locations are shown to the robot one after another, the whole human trajectory (all of the person positions) is segmented into sub-trajectories, where each sub-trajectory illustrates the set of person positions beginning at location A and ending at the subsequent location B shown by the guide-person. In particular the first sub-trajectory records the person positions from the beginning of the scenario to the first location, while the last one stores the person trajectory from the last location to the end of the guided tour. The whole graph G_{NoEdge}, namely the set of nodes N_i, is divided into graph layers G_{Layer}, on which the corresponding sets of person positions from sub-trajectories are stored. In other words, the graph G_{NoEdge} can be regarded as *graph union* or *graph join* of all graph layers G_{Layer} [21], since no edges have been connected between the nodes yet. Therefore, each node N_i of the graph G_{NoEdge} can be assigned to a corresponding layer $G_{Layer(N_i)}$[3], and all nodes belonging to a graph layer G_{Layer} represent a sub-trajectory of the guide-person. A virtual grid with a predefined resolution different from the resolution of the occupancy grid map is laid over the built map, and the number of nodes is limited to one per virtual grid cell.

The connection of graph nodes is created simply under the restriction of distance. Newly added nodes get connected by an edge to all neighbor nodes whose grid cell distance is smaller than a certain threshold. By this means, nodes lying on different graph layers might be connected, suppose they are near by each other; while there might be no edge between nodes on the same layer, when they are far from each other. A graph $G_{Multiple}$ can be therefore created with *graph join* of multiple layers G_{Layer} conditioned on the predefined distance restriction. The edge weights $\omega(N_i, N_j)$ are initially

[2]Abbreviation of Simultaneous Localization And Mapping.

[3]Note that a node with the same coordinate might be passed by the guide-person not only once. For such nodes they might belong to different graph layers.

set to the euclidean distance between the connected points N_i, N_j. The $A^\star$ search algorithm [22] is applied to find the least-cost path if it exists. With the initial weights the shortest distance path will be found.

However, the shortest path might be not human-like, as some details from the human trajectory could be ignored by $A^\star$. Therefore, the timestamp $ts(N_i)$ memorising when node N_i is added into the graph is taken into account. If the timestamp difference between the two selected nodes is greater than a defined threshold Θ_t, the initial edge weight will be multiplied by a penalty factor α_t, in that a long time span between two nodes may indicate that they are not neighbors on the original human path.

In addition, frequent changes between different human sub-trajectories should be prevented. The robot ought to follow one sub-trajectory as long as possible, unless the switches are necessary for the robot to reach a goal. As described above, instead of building one graph recording all person positions, multiple graph layers are created (see Fig. 4). Edges connecting nodes from different layers are allowed. However, they will suffer from a multiplied penalty factor α_l on the base of the initial edge weights. This, again, makes graph layer changes rare, although not impossible.

Altogether, the path calculated by $A^\star$ is a compromise between the shortest distance way from A to B and an imitation of human behavior which the robot watches during the *Home-Tour*. Thus, the edge weight $\omega(N_i, N_j)$ between two nodes N_i, N_j is defined as follows:

$$\omega(N_i, N_j) = p_t(N_i, N_j) \cdot p_l(N_i, N_j) \cdot \left|\left|\overrightarrow{N_i N_j}\right|\right| \tag{1}$$

where

$$p_t(N_i, N_j) = \begin{cases} \alpha_t & \text{if } |ts(N_i) - ts(N_j)| > \Theta_t \\ 1 & \text{otherwise} \end{cases} \tag{2}$$

and

$$p_l(N_i, N_j) = \begin{cases} \alpha_l & \text{if } G_{Layer(N_i)} \neq G_{Layer(N_j)} \\ 1 & \text{otherwise} \end{cases} \tag{3}$$

Note that $\alpha_t > 1$ and $\alpha_l > 1$. In section V we will show that the proposed approach is a trade-off allowing the robot both to benefit from qualities of human navigation and to make decisions for graph layer switches autonomously.

B. Replanning

A precise plan calculated from a static environment representation may be inappropriate to be executed because of dynamic objects, such as humans or moved furniture. Applying the laser sensor, ND [9] can avoid obstacles in the way to subgoals. Whenever a subgoal which may be near to a dynamic obstacle cannot be reached within a certain time period, the robot tries to go to the next subgoals. Thus, ND is expected to overcome temporal obstacles, such as passersby, or objects obstructing the path not completely, and drive the robot to the goal using the current plan.

If the application of the local approach (ND) is not successful, the current plan is marked invalid and obstacles between the robot and the subgoal are detected in the current

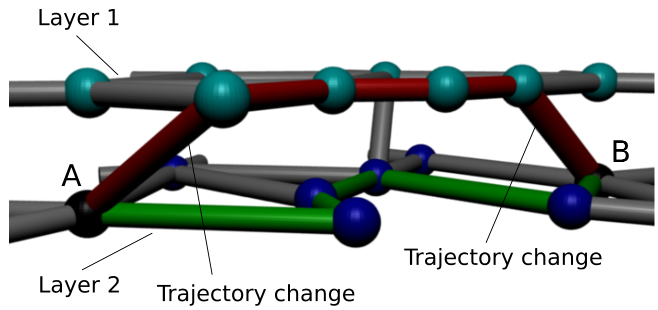


Fig. 4. Two layers created from the corresponding human sub-trajectories. Light blue nodes belong to layer 1, while dark blue nodes belong to layer 2 including the two black nodes A and B. Two possible paths from A to B are depicted with red and green edges. The graph search algorithm would prefer the green path, since the red one contains two expensive layer (sub-trajectory) changes.

occupancy grid map. For this purpose, grid cells contained in an ellipse covering the area between the robot and the subgoal are inspected, as Fig. 5 illustrates. If the percentage of occupied cells in this area exceeds a threshold, the object obstructing the current plan is confirmed and replanning is triggered. Considering the failure of ND and the detected obstacle between the robot and the next subgoal, this subgoal is marked as an unreachable node N_{ur}. Given the radius of the robot, the neighbors of the node inside this radius are marked as well. $A^\star$-search is then repeating, ignoring the invalid nodes. If no obstacle can be detected in the path found by $A^\star$ but ND still fails, NF1 is used for path planning as the last resort.

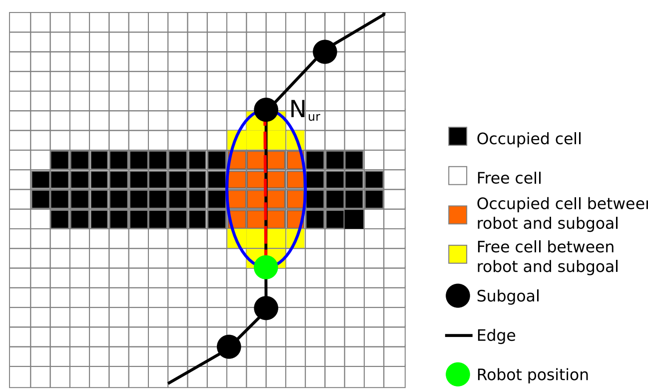


Fig. 5. Finding dynamic obstacles using the gridmap. Free grid cells are depicted in white, occupied cells in black. An obstacle is detected, since the occupied cells (orange) outweigh the free cells (yellow) in the area between the robot and the subgoal.

Furthermore, NF1 can be seen as a general fallback strategy whenever $A^\star$ cannot find a path from the built graph $G_{Multiple}$ discussed above. This is the case if either no connection within the graph between the start and the target node can be found, or the robot position is too far away from any node. The second case mainly occurs when the robot is manually placed at a location where it has not been before or it explored autonomously. While the robot is following the NF1 path, $A^\star$ constantly tries to find the path switching back to the preferred behavior computed from the graph $G_{Multiple}$ as soon as possible.

In this manner, the robot is expected to benefit from both the map-based path planning and navigation strategies of the guide-person, so as to solve the planning problem posed at the beginning of section I effectively.

V. Experiments

The proposed approach has been evaluated with **BIRON II** in a real environment resembling an apartment with real subjects. Fig. 6 illustrates the grid map of the environment built after the guided tour and the start as well as the end position of the robot. The gray-scale indicates the probability

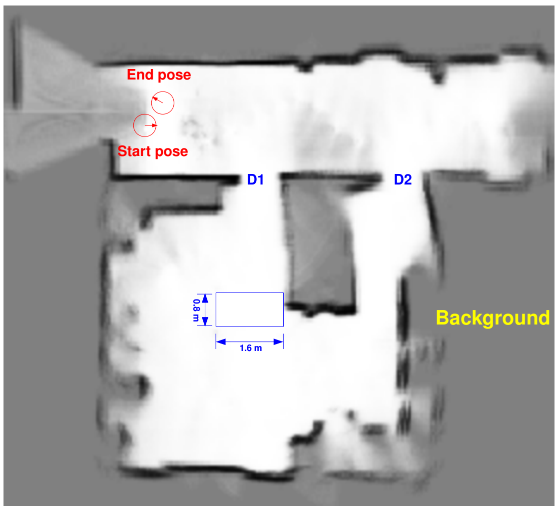


Fig. 6. Occupancy grid map of the environment. The both open door D1 and D2 during the *Home-Tour*, a table whose four legs can be perceived by the robot and the start as well as end pose of the robot are depicted in the map. The background indicating the unperceivable areas of the robot is marked as well.

whether the grids are occupied by obstacles perceived from laser sensor. White corresponds to free with high certainty, while black implies occupied with high certainty. The possiblity of the unperceivable grid cells, such as the background, is assigned 0.5 implying the maximum uncertainty. As the guide-person had to be regarded as an obstacle during the whole *Home-Tour*, the occupied area of the person was defined with the multi-modal person detection and tracking system mentioned in section III and excluded from the laser plane. A table about $1.60m \times 0.80m \times 0.72m$ is depicted with a blue rectangle. As the laser range finder can perceive a plane in front of the robot at a certain height, only the four legs of the table can be *seen* by the robot. The both doors D1 and D2 were opened, while the robot was following the guide-person. The multiple graph layers presented with different colors were created, when different locations were shown to the robot (see Fig. 7.). Comparing with the built *Navigation Graph* presented in [11] and [12] more details about the person movement and the environment representation are recorded. Map-based path planning, such as the NF1 approach, as well as the replanning process dealing with the dynamic objects that may obstruct the current plan are therefore feasible to be integrated into the proposed graph-based approach. As discussed in section IV-A, each layer stores the person positions from one location to another, if it exists. The green sub-trajectory provides the movement information of the guide-person from location A to B, while the graph layer with violet illustrates the set of the person positions from D to the end position of the guided tour, as no subsequent location is shown to the robot. Notice that edges connecting nodes from different layers are allowed to realize layer switches, when the sub-trajectories of the person interweave each other.

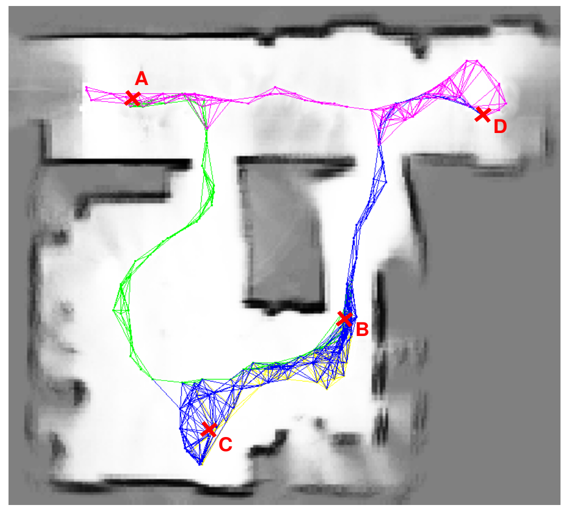


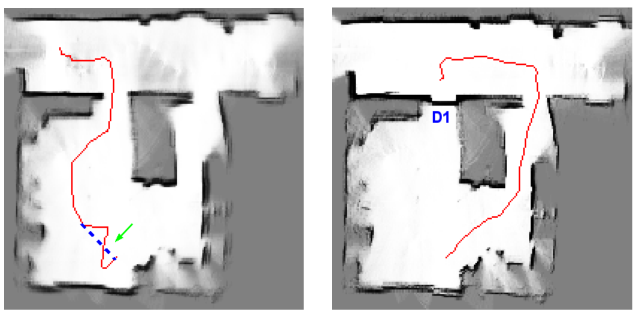
Fig. 7. Multiple graph layers created during the *Home-Tour*. Different colors illustrate different sub-trajectories of the guide-person from one location to another, as described in section IV-A.

The route of the guide-person began from A, via B, C, D, and ended in proximity to the start position A. The region between A and D corresponds to the corridor of the real environment. Since the robot followed the person and travelled through some places of the new environment only once, the uncertain (gray) areas, such as the corridor between the both doors could be found in Fig. 6. Those cluttered gray points in front of the robot at its start pose were created before the guide-person had been detected with the person tracking system. Once the guide-person was identified, no such points were added into the grid map during the subsequent guided tour.

After perceiving the new surroundings the robot would be expected to reach a goal set in the free space of the already visited environment. That is also a precondition for a domestic service robot. In the experiment the robot was required to return to place C. The path for the robot to the goal is computed using the approach described in section IV-A and illustrated in Fig. 8(a). Since no graph layer containing the sub-trajectory of the guide-person from A to C is available, layer change is necessary for the robot to find a path. In order to compare the resulting path from our approach with that from NF1, the both trajectories are depicted with red and green respectively in Fig. 8(b). As could be anticipated, the map-based approach NF1 found out

(a) (b)

Fig. 8. The computed path from the graph-based approach presented in section IV-A and the map-based approach NF1. a) shows the found red path from the built graph. Note that the path with a small detour results from the edge weights defined in formula 1. The green path from NF1 crossing the table is illustrated in b) compared with the red path containing navigation strategies of the guide-person.



(a) (b)

Fig. 9. Comparison of the planned and replanned path. a) shows the planned path on the base of the built map during the *Home-Tour*. Tuning edge weights defined in formula 1 the blue dashed shortcut would be an alternative neglecting potential obstacles in the region shown by the green arrow. Once the new obstacle D1 had been added in the grid map and detected by the robot, replanning was started. The resulting path is illustrated in b).

a shortest way to the goal without the consideration of the *unperceivable* table, while the red path contains the strategy of the guide-person.

Afterwards the robot tried to reach the subgoals on the way which the guide-person had passed by to C, i.e. the red path illustrated in Fig. 9(a). However, the door D1 lying on the robot path was closed *purposely*. Once the local approach ND was not successful and the robot was *aware of* the new obstacle D1 in front which obstructed the path, as discussed in section IV-B, replanning was started. The replanned path computed from the graph and the closed door D1 added into the map are shown in Fig. 9(b). Note that the uncertain (gray) regions between the both doors are relatively certain (white) now, compared with Fig. 6, since the robot has passed by this area again. When the robot has reached C successfully applying the replanned path, it was commanded to go back to A. In principle, tuning edge weights defined in formula 1 a shortcut depicted with blue dashed in Fig. 9(a) is possible. However, potential hazards that can be avoided with the red path might exist in the area shown by the green arrow.

Except for comparing the paths planned from the map-based and our approach in an environment partially perceived by the robot, as shown in Fig. 8(b), we are interested in the plans resulting from a *fair* situation as well, i.e. the obstacles on the way to the set goal can be perceived by the robot entirely. Therefore the robot position with orientation was marked, when the robot went back to A, and the same goal B was sent to the robot applying the path from our approach and NF1. The result is illustrated in Fig. 10. As NF1 tries to find a shortest path, the green trajectory leads the robot near to obstacles, such as the wall and the door D2. In contrast, more free space and smooth obstacles avoidance have been considered by the guide-person for safely planning, as the red trajectory shows, especially in the corridor and through the door D2. The small mapped legs of the table depicted in Fig. 6 are shown with the blue arrows as well.

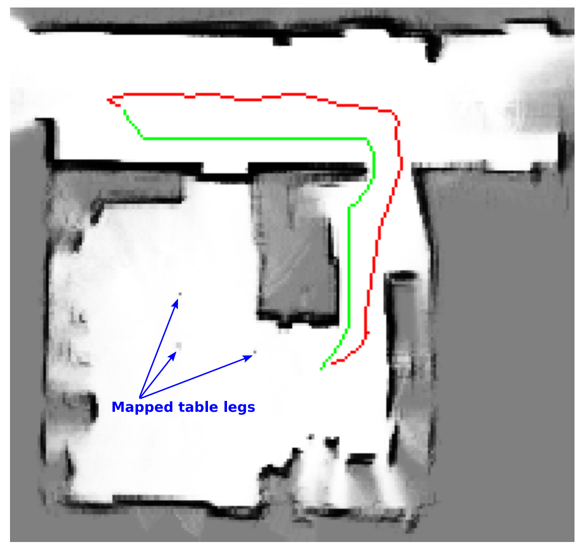


Fig. 10. Comparison of the paths computed from NF1 and our approach for the robot from A to B, given in Fig. 7. The green path from NF1 is close to obstacles, while the red path reveals navigation strategies of the guide-person. The mapped table legs are shown with the blue arrows as well.

VI. Conclusion

This paper presented a path planning approach on the basis of a graph with multiple layers, representing movement trajectories of guiding persons acquired during the so-called *Home-Tour*. In contrast with the approaches based purely on robot sensors, the human-like paths would be beneficial for the robot considering navigation capabilities of human beings, particularly when obstacles could not be perceived by robot sensors. In addition, the trajectory of the guide-person should be expected to result in collision free paths, in that those paths implicitly contain human strategies for obstacle avoidance, as well as human understanding of spacial context. For the graph-based approach, efficiency, effectivity and keeping the trajectory originally from the guide-person have to be considered synthetically to compute a robot path. Therefore, we have defined edge weights with multiple factors, which take into account distance to the goal, timestamp along the person trajectory and switches of graph layers. Since the ultimate goal for path planning is to find a path from any location to any other location in the environment, the map-based approach NF1 has been integrated into the current system as a supplement to overcome the limitation of the graph-based approach, i.e. it is almost impossible for the trajectories of the guide-person to cover any free space of the environment. If unexpected dynamic obstacles obstruct the path, a cascade of replanning processes will be triggerd trying to find an alternative. Comparing with the works of Zender et al. [11] and Topp [12] we combine the advantages of the graph-based search approach and the map-based navigation (NF1) with the help of the occupancy grid environment representation. This integrated approach enables our robot system to more safely, while still efficiently navigate in its environment.

Considering that the planned robot paths are calculated in terms of the movement trajectory from the guide-person, a practical and efficient solve might be obtained through analyzing and optimizing the person paths. Trajectories interwoven with or closed to each other are generally caused by repeated movements of the person through the same areas. Merging those trajectories can refine the graph and make the path planning and replanning process more efficient. Although our approach is on the basis of the meaningful person trajectories, unnecessary detours obtained from the guide-person might be difficult to avoid in practice. Through a comparison of the person movement with the perceived spatial context of the environment, the robot would be expected to clarify the necessity of the current person behavior in an inquiring manner.

Besides, the robot should carry out assignments on the built map for next time. In other words, the robot has to be able to localize itself on the known environment representation. Hence, a particle filter based approach from the MRPT library [19] for robot global localization [23] has been integrated into the current system as well.

Notice that only the small table legs shown with the blue arrows in Fig. 10 can be perceived by the robot and recorded on the map. Adopting navigation strategies of the guide-person such an unperceivable obstacle will not impede the robot to perform navigation tasks though, in a dynamic environment such a movable table may be placed on the planned path for the robot and become a potential hazard. In fact, two dimensional perceptual space of a fixed mounted laser is not sufficient to ensure safe navigation in an unconstrained environment. A motion generation taking 3D perception into account is therefore meaningful and necessary. For the future we are going to integrate a 3D Time-of-Flight camera on the robot platform as an extension of the 2D laser sensor for a full 3D motion generation from our previous work [24].

Acknowledgments

This work has been supported by the Collaborative Research Centre 673 *Alignment in Communication*, funded by the German Research Foundation (DFG), and the European Community's Seventh Framework Programme [FP7/2007-2013] under grant agreement No. 215181, *CogX*.

References

[1] J. A. Meyer and D. Filliat, "Map-based navigation in mobile robots II: A review of map-learning and path-planning strategies," in *Cognitive Systems Research*, R. Miikkulainen, Ed. Elsevier, 2003, vol. 4, pp. 283–317.

[2] K. Craik, *The Nature of Explanation*. Cambridge University Press, 1943.

[3] J.-C. Latombe, *Robot motion planning*. Dordrecht, Netherlands: Kluwer Academic Publishers, 1991.

[4] M. J. Schoppers, "Universal plans for reactive robots in unpredictable environments," in *Intl. Joint Conf. on Artificial Intelligence*, 1987, pp. 1039–1046.

[5] M. Hauskrecht, "Value-function approximations for partially observable markov decision processes," *Journal of Artificial Intelligence Research*, vol. 13, pp. 33–94, 2000.

[6] N. L. Zhang and W. Zhang, "Speeding up the convergence of value iteration in partially observable markov decision processes," *Journal of Artificial Intelligence Research*, vol. 14, pp. 29–51, 2001.

[7] D. Fox, W. Burgard, and S. Thrun, "The dynamic window approach to collision avoidance," *IEEE Robotics and Automation Magazine*, vol. 4, no. 1, pp. 23–33, 1997.

[8] J. Borenstein and Y. Koren, "The vector field histogram-fast obstacle avoidance for mobile robots," *IEEE Transactions on Robotics and Automation*, vol. 7, no. 3, pp. 278–288, 1991.

[9] J. Minguez, "Nearness diagram (nd) navigation: Collision avoidance in troublesome scenarios," *IEEE Transactions on Robotics and Automation*, vol. 20, no. 1, pp. 45–59, 2004.

[10] R. Gockley, J. Forlizzi, and R. Simmons, "Natural person-following behavior for social robots," in *Proceedings of Human-Robot Interaction*, March, 2007.

[11] H. Zender, Ó. M. Mozos, P. Jensfelt, G.-J. M. Kruijff, and W. Burgard, "Conceptual spatial representations for indoor mobile robots," *Robotics and Autonomous Systems*, vol. 56, no. 6, pp. 493–502, 2008.

[12] E. A. Topp, "Human-robot interaction and mapping with a service robot: Human augmented mapping," Ph.D. dissertation, School of Computer Science and Communication, Royal Institute of Technology, Stockholm, Sweden, 2008.

[13] A. Haasch, S. Hohenner, S. Hüwel, M. Kleinehagenbrock, S. Lang, I. Toptsis, G. Fink, J. Fritsch, B. Wrede, and G. Sagerer, "Biron – the bielefeld robot companion," in *Proceedings of the Workshop on Advances in Service Robotics*, Stuttgart, Germany, 2004, pp. 27–32.

[14] F. Yuan, M. Hanheide, and G. Sagerer, "Spatial context-aware person-following for a domestic robot," in *International Workshop on Cognition for Technical Systems*, Munich, Germany, 2008.

[15] S. Lang, M. Kleinehagenbrock, S. Hohenner, J. Fritsch, G. A. Fink, and G. Sagerer, "Providing the basis for human-robot-interaction: A multi-modal attention system for a mobile robot," in *Intl. Conf. on Multimodal Interfaces*, 2003.

[16] H. Durrant-Whyte and T. Bailey, "Simultaneous localisation and mapping (slam): Part I the essential algorithms," *Robotics and Automation Magazine*, vol. 13, pp. 99–110, 2006.

[17] A. Doucet, N. de Freitas, K. P. Murphy, and S. J. Russell, "Rao-blackwellised particle filtering for dynamic bayesian networks," in *Conf. on Uncertainty in Artificial Intelligence*, 2000, pp. 176–183.

[18] G. Grisetti, C. Stachniss, and W. Burgard, "Improved techniques for grid mapping with rao-blackwellized particle filters," *Robotics and Autonomous Systems*, vol. 23, pp. 34–46, Feb. 2007.

[19] J.-L. Blanco, "The mobile robot programming toolkit (mrpt) website," 2008. [Online]. Available: http://babel.isa.uma.es/mrpt/

[20] R. Philippsen, "Motion planning and obstacle avoidance for mobile robots in highly cluttered dynamic environments," Ph.D. dissertation, École Polytechnique Fédérale de Lausanne, 2004.

[21] F. Harary, *Graph Theory*. Reading, Mass: Addison-Wesley, 1969.

[22] P. E. Hart, N. J. Nilsson, and B. Raphael, "A formal basis for the heuristic determination of minimum cost paths," *IEEE Transactions on Systems Science and Cybernetics*, vol. 4, no. 2, pp. 100–107, 1968.

[23] S. Thrun, D. Fox, W. Burgard, and F. Dellaert, "Robust monte carlo localization for mobile robots," in *Artificial Intelligence*. Elsevier, 2001, vol. 128, pp. 99–141.

[24] F. Yuan, A. Swadzba, R. Philippsen, O. Engin, M. Hanheide, and S. Wachsmuth, "Laser-based navigation enhanced with 3d time-of-flight data," in *Intl. Conf. on Robotics and Automation*, Kobe, Japan, 2009, pp. 2844–2850.