Master Thesis Proposal

A Global Trajectory Planner for Mobile Robots

Jens Steinhauser
e0826623@student.tuwien.ac.at

Supervisor: Dipl.-Ing. Dr.techn. Markus Bader

May 4, 2022

1 Motivation & Problem Statement

The proposed thesis will present a global path planning which takes the physical constraints of the underlying vehicle into account. In order to navigate a mobile robot from one pose to another, various software components are used that work together to solve the problem. Typically these software components work on the basis of *sense*, *think*, and *act*. The "sensing" part is about localizing the robot in the environment, "thinking" in this context means to create a plan to the goal, and "acting" is to execute the calculated plan.

Motion planning encompasses the thinking and the acting stage of autonomous navigation, and can be broken down into the three parts *routing*, *global planning*, and *local planning*. These parts work on different levels and time frames:

- Routing establishes the next goal point for the robot to move to. In its simplest form this part could be the translation of direct user input to the target pose, but could also be a more sophisticated routing algorithm that determines which robot has to go where. This part of the software typically runs in intervals of multiple seconds.
- Global planning is about finding a path connecting the start and the goal poses on the map of the environment. The map contains the static obstacles known at the time, and the global planner computes a series of poses from the start to the goal to navigate through the environment and around static obstacles without collision. Typical algorithms used to search through the graph built from the map are the Dijkstra-algorithm or an optimization of it (like A*), or rapidly exploring random trees (RRT). The global planner runs with a frequency in the order of seconds and is re-triggered when needed.

- Local planning takes the global path from the previous step as input and creates steering commands for the robot's driving wheels to follow the global path as close as possible. Additionally, the local planer has to react or drive around dynamic obstacles if they appear in the robot's path. To keep the robot on track, the local planner runs with a higher frequency than the other parts, typically ranging from 10 to 50 Hz, and takes real-time sensor data like wheel odometry and localization data into account.

The specific construction of a robot limits the trajectories it can follow. For example, the selection of the robot's motors influence the maximum speed, but also how well it can counteract inertia, which in itself is dependent on the robot's construction; or the arrangement of the wheels dictate the steering behaviour of the robot.

Global planning algorithms typically take the robot's pose and its dimensions into account when searching for a plan, but not the limited control space that stems from the robot's construction [5], as this would inflate the search space and slow down the planning. Instead, the limitations of the robot are handled in the local planner where they limit the permissible steering commands, with which the local planner has to try to stick to the given global path as close as possible. This can lead to the situation where the global planner produces an input for the local planner that it can not follow, which leads to problems, in particular when navigating in narrow spaces.

2 Expected Results

As described in the previous section, current global planners do not take into account the detailed physics dynamics of the underlying vehicle, and this causes issues for the local planner to follow the global path. However, the increase in mobile computing power, combined with a refined pruning strategy for the search space, make it worthwhile to attempt the development of a global planner that incorporates the physical limitations to generate better global paths.

In particular, a global planner for differential wheeled robot shall be developed that can integrate into the navigation stack of ROS2 [6]. This planner shall be able to take velocity and acceleration limits of an underlying vehicle into account and a comparison to the existing ROS2 planner shall be done. The new global trajectory planner will take physical limits of an underlying vehicle into account and will be integrated into ROS2. Finally the thesis will present a detailed study of the performance gain by comparing the developed planner to state of the art planners implemented in the ROS2 navigation system.

3 State of the Art

The idea of including the vehicle's characteristics in the global planning stage is not new, [1] mentions it under the term "Global Time-Optimal Trajectory Planning", but also describes the problems that arise from the big search space and the resulting effort needed to process it.

A software collection covering all areas of robotics-related research can nowadays be found in the *Robot Operating System* (ROS) [7] project. It includes open-source implementations of state-of-the-art algorithms for all kinds of robotics applications like mapping, localization, perception, and also navigation. ROS' navigation stack (called "move_base" in ROS1, the older version, or "nav2" [6] in the more recent ROS2), contains a number of Dijkstra and A* based global planners, and a number of local planners. The navigation stack has a plug-in interface that allows the easy addition of new planners implementing new ideas, and a number of planner plugins for the ROS navigation stacks exist outside of the ROS project, [2] gives a good overview.

The community put effort into optimizing the local planners to output a good trajectory for a given vehicle, starting with the classic "Dynamic Window Approach" planner [3], the "Timed-Elastic-Band" planner [8], and also from our university [9], but not for global planners.

4 Methodological Approach

The steps planned to do this thesis are as follows:

- Literature research
  An extensive survey shall be conducted to get an overview on the existing planning solutions for mobile robots and to deepen the understanding of the problem statement.

- Implementation
  The implementation of an improved planner as described in the section 2 shall be done. The implementation shall be supported by a simulation in the Gazebo simulator [4].

- Evaluation on simulated and real hardware
  The newly developed planner shall be put into action on one of the robots available in the institute's laboratory.

- Comparison
  The driving behaviour of a robot using the planner that was developed for this thesis shall be compared to the behaviour of it driving with existing planners currently available in the ROS navigation stack.

5 Relevance to the Curricula of Computer Engineering

The curricula of Computer Engineering contains the lecture "Mobile Robotik" that covers topics mentioned in this proposal like localization, global and local path planning. In addition, as robotics is an interdisciplinary field that touches and integrates many parts of mechanical, electrical, and software engineering, other lectures of the curricula also provide useful input in the fields of algorithms, optimization, control theory, and automation:

183.660	Mobile Robotics
186.814	Algorithmics
186.112	Heuristic Optimization Techniques
182.763	Stochastic Foundations of Cyber-Physical Systems
376.059	Control systems 1
376.060	Control systems 2
376.058	Optimization

References

[1] Howie Choset, Kevin M. Lynch, Seth Hutchinson, George Kantor, Wolfram Burgard, Lydia Kavraki, and Sebastian Thrun. *Principles of Robot Motion: Theory, Algorithms, and Implementations.* MIT Press, 2005. ISBN 0262033275.

[2] Alexandros Filotheou, Emmanouil Tsardoulias, Antonis Dimitriou, Andreas Symeonidis, and Loukas Petrou. Quantitative and Qualitative Evaluation of ROS-Enabled Local and Global Planners in 2D Static Environments. *Journal of Intelligent & Robotic Systems*, 98:567–601, 2020.

[3] D. Fox, W. Burgard, and S. Thrun. The dynamic window approach to collision avoidance. *IEEE Robotics Automation Magazine*, 4(1):23–33, 1997. doi: 10.1109/100.580977.

[4] N. Koenig and A. Howard. Design and use paradigms for gazebo, an open-source multi-robot simulator. In *2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) (IEEE Cat. No.04CH37566)*, volume 3, page 2149–2154, Sendai, Japan, 2004. IEEE. doi: 10.1109/IROS.2004.1389727.

[5] Steven M. LaValle. *Planning Algorithms.* Cambridge University Press, 2006. ISBN 978-0521862059.

[6] Steven Macenski, Francisco Martin, Ruffin White, and Jonatan Ginés Clavero. The Marathon 2: A Navigation System. In *2020*

IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2020.

[7] Morgan Quigley, Brian Gerkey, Ken Conley, Josh Faust, Tully Foote, Jeremy Leibs, Eric Berger, Rob Wheeler, and Andrew Ng. Ros: an open-source robot operating system. In *Proceedings of the IEEE Intl. Conf. on Robotics and Automation (ICRA) Workshop on Open Source Robotics*, Kobe, Japan, May 2009.

[8] Christoph Rösmann, Frank Hoffmann, and Torsten Bertram. Integrated online trajectory planning and optimization in distinctive topologies. *Robotics and Autonomous Systems*, 88:142–153, 2017.

[9] Horatiu George Todoran. Optimal local path-planning and control for mobile robotics. Master's thesis, Vienna University of Technology, 2018.