Distributed Mapping and Path Planning

Benjamin Schnieders

September 03, 2015

Abstract

Mapping the environment and planning paths is a substantial part of maintaining the ability to move for mobile robots. Robots typically obtain map data through a technique called simultaneous localization and mapping (SLAM), and use path planning algorithms on the generated map to find routes to arbitrary locations. This research proposal evaluates the possibilities and advantages of extending the well-established, mostly single-agent based SLAM and path planning algorithms towards the domain of multi-agent systems.

By taking advantage of increased overall computational power and storage capabilities of a multi-agent system, larger or more detailed maps can be stored than possible on a single robot. Multiple agents can cooperate in planning a path for a single agent in order to speed up the pathfinding procedure. Spreading the map over multiple agents requires a distributed path planning procedure in order to find a path without assembling a monolithic map.

Systems of multiple agents are less susceptible to single points of failure by adding additional redundancy, thus enhancing the overall robustness of the system. Incorporating inputs originating from different types of sensors to a navigable map, a diverse robotic setup can be achieved while retaining robustness and redundancy.

Practical applications of distributed slam emerge whenever a single-robot setup is prone to complete failure, i.e., where the robot is endangered by its environment and cannot be replaced easily. In these situations, which include extraplanetary rovers or surgical microbots, multiple, cooperating robots may be used instead. Systems of multiple robots, especially those designed to work as a robot swarm, such as unmanned aerial vehicles looking for survivors in a disaster aftermath, benefit from distributed SLAM naturally, as it reduces their individual memory usage for navigation, allowing more important data to be stored or lager regions to be explored.

Keywords: SLAM, autonomous navigation, multi-agent system, distributed pathfinding, distributed map storage, mobile robotics

1 Introduction

This Ph.D. research is aimed at achieving robotic systems of higher robustness by distributing the navigation procedures, which include exploration, mapping, and path planning among multiple robots. This introductory section provides an insight to the significance and the background of the field of research.

1.1 Significance

Mobile robotics is a highly active field of research. Competitions such as the RoboCup [36] or DARPA Grand Challenge [34] demonstrate the state-of-the-art AI techniques used. As robotic movement, compared to the accuracy and robustness of human motion, is still in its infancy, mobile robots are mostly used in areas too dangerous or otherwise inaccessible to humans, amongst which are space exploration [71][49] or robotic minesweeping [40]. Due to the nature of these tasks assigned to robots, robustness and autonomy are of greatest significance, as human intervention is often physically impossible or costs infeasibly high effort [73].

Research in the area is heavily driven by governmental institutions such as NASA, ESA or DARPA, but also, to an increasing amount, by companies such as the publicly held Google, Inc [23].

1.1.1 Practical Applications

Practical applications of Distributed Mapping and Path Planning range from planetary exploration to disaster recovery assistance.

Every system of cooperating mobile robots can benefit from distributed mapping and planning, as it decreases their individual latency until a path is planned, and reduces the overall amount of memory dedicated for map storage. Robots designated to be used in a swarm are typically designed to be small and cost-efficient [45][52], which prevents usage of high-performance onboard computer systems. Less advanced onboard computers demand the usage of efficient, possibly distributed and parallelized algorithms.

Aforementioned space exploration is typically performed by single robot systems, such as planetary rovers. Getting stuck in a natural environments is a serious issue especially for wheeled or tracked robots. A single robot often has no means to escape a stuck situation, forming a total loss. Using distributed mapping and planning, a multi-robot setup may be used, in which a failure of one robot does not constitute a loss of the complete system.

Having limited lifting capacities, aerial robots are designed as lightweight as possible [9]. By reducing the computational expenditure for navigation using distributed mapping, more energy-efficient CPUs can be used, which in turn reduces weight formerly dedicated for heatsinks; even battery life may be prolonged. By spreading the navigable map over all robots in the system, each robot only has to store a small part of the map. The gained free memory can then be used by other applications, like image processing.

For example, a swarm of micro-drones could be used to map disaster regions and use image recognition and classification techniques to find survivors. Improved navigation accuracy, increased battery life and thus flight time and an increased amount of free memory to perform the necessary, potentially life-saving non-navigatory tasks greatly enhance the performance of the whole system.

1.2 Background

Simultaneous localization and mapping (SLAM) and path planning are of great importance for mobile robots. During the last decades, these problems have been extensively researched and the generated results are of sufficient quality for many applications [42]. The main challenge appears to be the restricted computational capabilities of robots, lacking large amounts of memory and high-performance CPUs [15]. This challenge is commonly faced by producing more efficient algorithms for specific robot designs, and reducing the memory footprint by storing only the bare minimum of information needed to navigate an environment [63][61].

Another possibility to overcome the computational limitations of a single robot is to use multiple robots [53]. A multi-robot setup does not suffer as much from single points of failure as a single robot setup. As long as inter-robot communication is operative, tasks may be diverted to functioning robots. Exploration of unknown areas can be significantly sped up by employing multiple robots simultaneously [25]. Mapping accuracy can be increased by measuring distances between individual robots, using them as artificial landmarks [1] or generating otherwise unavailable loop closures [35].

Distributing map information among multiple robots, a larger map area can be held in memory at all times. Alternatively, more accurate maps, i.e., maps with a higher spatial resolution, or maps with additional information like laser-obtained surface texture [65] or ultrasonic reflectivity [47] can be saved, and accessed by inter-robot communication. This additional data can be used in planning, such as planning lower velocities on a rough surface or lower accelerations on a potentially slippery, even surface. Assuming excessive utilization of the shared robot map memory, a single robot can not save the whole map by itself any more. Planning a path beyond the region known to a single robot thus requires efficient multi-agent path planning algorithms.

Equipping multiple robots with all the same type of sensors is highly redundant. While redundancy enhances robustness, a setup with different sensor types is more versatile, as it can record different information. Camera data and laser or ultrasonic distance measurements may be combined to a distributed map. Different sensor types may be used to produce a hierarchical map, with regions of higher and lower information density, obtained from the different available sensor types.

2 Research Scope

This Ph.D. research is focused on extending the established mapping and path planning algorithms to utilize multiple robots. The core research can be defined through the following research questions.

2.1 Research Questions

Multi-agent exploration and mapping commonly uses technologies such as map-fusing to incorporate single agent maps into an overall terrain map [50][37].

Trying to circumvent a monolithic map assembly raises the first research question:

> *How can common SLAM algorithms be enhanced to produce effectively accessible distributed maps?*

To be able to plan on a distributed map, a likewise distributed, efficient planning algorithm is called for:

> *How can multiple agents efficiently cooperate in planning a path for a single agent?*

Compared to single robot setups, the multi-robot solution will have to be more robust to failure, in order to provide an improvement. This leads to the following research question:

> *How can multi-robot mapping and planning technologies be designed to be more robust than their single-agent counterparts, regarding events such as technical failures or connection losses?*

Multi-agent communication is a deeply researched field. Nevertheless, domain specific problems such as identifying the robots and possibly limited physical communication abilities pose the following research question:

> *How can mobile robots efficiently communicate their estimated positions and recorded maps to one another?*

In order to be called superior, the multi-robot setup will have to be more versatile than a single robot setup. Enabled by the increased amount of possible storage, the ability to store and use map features obtained from different sensor types will highly increase the versatility of the setup. This leads to the final research question:

> *Can sensory data of different sensor types be combined to form a map?*

3 State of Research

Single robotic movement, mapping and planning are widely researched topics, providing a good foundation for further research. This section provides a brief review of all the techniques necessary for robotic movement.

3.1 Movement and Path Planning

In order to move to a specified location, a robot or any situated agent requires a step by step movement plan, containing navigation instructions on how to reach the goal [16]. This plan is typically achieved by pathfinding, a common AI technique to find the shortest route between two nodes on a graph [53]. This search graph is modeled in a specific way such that the nodes correspond to actual locations in the physical environment, and can be obtained from a map representation.

Generally, an agent can only move on the surface of the mapped terrain, with the exception of amphibian/submersible vehicles or flying robots. The terrain surface is then converted to a two dimensional search graph. In this search graph, distances between corresponding physical locations to any two connected nodes are incorporated as edge costs. More terrain information, such as height or slopes may be included in the graph nodes. Subsequently, search algorithms are used to find a path between the start and the goal node.

A returned path consists of a list of nodes that have to be navigated in order to reach the goal. Uninformed search algorithms such as (depth-limited) depth first search or Dijkstra's algorithm [19] can be used, however, more commonly, informed search algorithms such as A* [30] and IDA* [38] are used. These algorithms make use of a heuristic evaluation function, to change the order in which nodes are expanded to a likely more efficient one.

After the path has been returned, the nodes are re-translated into the corresponding physical locations. The lowest-cost edges between consecutive nodes on the path are translated into movement controls for the robot, providing an optimal traversal of the terrain. Pathfinding algorithms like A* and IDA* have been proven to be successful in computer gaming, producing movement plans for many agents in real time [44]. Besides finding shortest paths in spatial representations, path finding algorithms can be used to plan action sequences producing physically plausible paths [54].

3.2 SLAM

Mobile robots in an unknown environment face a problem of recursive nature; in order to properly locate themselves, a map representation is required. However, to build a map of the environment, the current position of the robot taking measurements is required as well. To overcome this mutual dependency, simultaneous localization and mapping iteratively reduces the error in both the estimated position as well as the generated map so far [3]. Sensory and odometry noise further complicate this challenging problem. While certain odometry errors can be corrected on-line, a baseline noise persists [5]. Common sensor types include ultrasonic sensors [64] and microwave radar sensors [21][27]. Due to developments in the field of semiconductor diodes, a number of SLAM algorithms based on scanning laser range finders have been developed [46][57][8]. For robots equipped with camera sensors, monocular SLAM algorithms exist [14][59]. Special algorithms can make use of depth information retrieved from stereoscopic camera systems [32].

Providing an affordable RGB-D sensor, the Microsoft Kinect™ has been successfully used for mapping approaches [29]. Different sensor inputs can be combined to incorporate more information into the generated map using an approach called sensor fusion. Ultrasonic and LIDAR [20] or LIDAR and monocular vision sensors can be combined [11], effectively increasing the amount of available mapping information. More recently, multi agent SLAM attracted research focus, generating maps obtained from multiple robot's sensory data while simultaneously estimating poses for all robots [70][2][18]. Landmarks visible from all robots may be used as mapping guidance or to generate loop closures [24][12]. Research concerning map distribution among all robots in the system to improve upon the limitations of map-merging techniques exists [17], however, the required improvements to path planning algorithms to make use of decentralized maps are not considered.

3.3 Distributed Pathfinding

The field of distributed pathfinding can be roughly subdivided into following research areas [62][22][51]:

1. Multi-Agent Pathfinding
2. Assisted Pathfinding
3. Parallel Computation
4. Purely Distributed Pathfinding

All these areas contrast typical pathfinding routines, for they are concerned with path planning in setups where the executing and planning agents are not necessarily the same. (1) Multi-Agent Pathfinding provides collision-free paths for multiple robots. (2) Computationally lightweight robots may be guided by sensor networks, in which case the computation is done on the network, not on the robot. (3) In order to increase the computation speed, the pathfinding algorithm may be parallelized and run on multiple processors the same time. (4) Purely distributed pathfinding aims to increase the computational capabilities of path-planning robots by harnessing the combined computational and storage resources of a multi-robot system.

3.3.1 Multi-Agent Pathfinding

A certain amount of research has been performed in the field of multi-agent pathfinding. Available works focus on planning collision-free multi-agent plans in abstract spaces [56][55], planned from a single agent's perspective. Algorithms used range from various improvements on A* and D* [28] to entirely newly developed algorithms such as MAPP [66]. More recent work also provides the robots with decentralized pathfinding [68][69], however, these algorithms only function under the assumption of perfect map knowledge.

3.3.2 Assisted Pathfinding

The area of assisted pathfinding enables agents with limited resources to navigate in environments too complex for them to process. Navigation is performed on sensor networks which are scattered over the environment. These sensors provide local paths in their surroundings, and, through inter sensor communication, can build a global path, which is sent to the navigating agents as a series of movement instructions [4]. While recent work provides distributed algorithms with reduced network overhead, the underlying assumption in assisted pathfinding is that sensor networks are of static nature [41].

3.3.3 Parallel Computation

Commonly available computer hardware features multiple processor cores onboard. To make use of these additional calculation resources, algorithms that are designed to run on single-core computers have to be extended or re-designed. Differentiating parallel pathfinding from purely distributed pathfinding, these algorithms typically exploit the hardware setup being able to concurrently access the whole map [31]. Even recent algorithms, designed to be run in a distributed manner, rely on the map information being global and accessible by each agent [48].

3.3.4 Purely Distributed Pathfinding

While parallel and assisted pathfinding make assumptions of the complete map being accessible to any processing node, or processing nodes being located in physical and map-space statically, purely distributed pathfinding solutions should function without these assumptions. Recent work [51] eliminates the assumption about all processing nodes having access to a globally shared map, however, this research is still based on the necessity of a static node network. A purely distributed pathfinding algorithm should be able to plan paths either optimally or to a desired quality with each computational node only being aware of a limited map region. As the layout of mobile robots is dynamic, the distributed map representation is prone to change, with each robot storing a map region physically closest to its own position.

3.4 Innovative Elements

Up to date, no purely distributed mapping and path planning algorithm exists. Existing research is either based on the assumptions of globally and completely accessible maps, or on static layouts of computing nodes. Works researching the combination of different sensory inputs rely on a static transformation being available between different sensors. Incorporating sensor data obtained from different sensors on different robots in a distributed way is a novel approach. While parallelized pathfinding algorithms have been developed, performing hierarchical, distributed path planning on distributed map data extends the current realm of research.

The novel elements can therefore be listed as:

- *Distributed Map Storage* for enhanced resolution and map sizes in dynamic robotic networks
- *Distributed Path Planning* efficiently calculating paths on a distributed map
- *Sensor Fusion* of sensors mounted on different robots into a distributed map

4 Approach

Research will begin with an in-depth analysis of the available exploration and mapping algorithms, efficient and robust distributed data storage as well as distributed agent path planning techniques.

Novel approaches will be developed to build a distributed, multi-robot exploration, mapping and path planning framework.

The developed approach can then be evaluated on Maastricht University's available robots, such as the Nao or TurtleBot robot platforms. Research comparing the performance of the presented approach to state-of-the-art techniques and exploring the fault tolerance and robustness will conclude the project.

The following subsections will provide detailed information on how the individual sub-problem are approached.

4.1 Distributed Map Storage

As storing a navigation map in a distributed manner is a novelty in the field, an innovative map storage system will be developed. Existing decentralized distributed storage systems, such as Chord [58] or Kademlia [43], will be evaluated regarding usability for dynamic, distributed map storage. The selected approach will then be extended in order to achieve a link between robot position and stored map segment. Incorporating the physical position of the map segments recorded into the hashing function for the distributed hash table, information can be retrieved and ordered by physical location. This allows quick local access, as a robot planning to a nearby location will hold the necessary map regions in its own memory. More distant map segments can be retrieved from the robot closest to the queried location in the hash table key space.

Fault tolerance can be achieved by adding a degree of redundancy to all agents [10]. The stored map data may be decomposed to multiple levels of detail before distribution, allowing to store more coarse map features wide-spread, and storing finer details more locally.

The finished distributed map storage system will be measuring against existing local storage solutions, comparing fault tolerance, data access speeds and available storage size. Simulations can show which communication bandwidth is required to efficiently lookup maps and communicate changes between agents, especially if agents need to swap map data due to physical location changes [39].

4.2 Purely Distributed Pathfinding

In order to develop a purely distributed pathfinding algorithm, existing hierarchical pathfinding algorithms [6] and parallel search algorithms [31] will be evaluated. Hierarchical path finding decomposes the map to different levels of accuracy, and refines the generated path incrementally from the coarser to the finer levels. Near-optimal results can be achieved with a reduced computational load [60]. By sharing a common coarse map representation, local path refinement may be spread over multiple agents.

The newly developed algorithm will thoroughly tested against comparable approaches. Assuming a static robot layout, the path qualities, measured as deviation from the optimal path, as well a planning times and network communication overhead, can be compared to sensor network pathfinding approaches, such as zePPeLIN [51]. For different types of setups, with varying number of agents, map decompositions and terrains, the resulting path quality and planning overhead can be compared to single-agent pathfinding results.

4.3 Combination of Sensory Input

To be able to combine different sensory inputs, corresponding landmarks have to be found in each input sequence. To efficiently recognize features among different types of sensors, obtained from mobile, physically separated robots on a distributed map, development of a novel algorithm is recommended. While typically, landmark detection requires a certain degree of domain knowledge and pre-processing [13], which is undesirable for processing of data from different origins. However, uninformed landmark detectors can achieve satisfactory results as well [67], and may be well suited for the task. To compensate for different resolutions and input dimensions, a multi-level wavelet decomposition can aid identifying features [33] at high accuracy even on low-detail inputs. Corresponding landmarks can subsequently be matched using the RANSAC [26] algorithm, which is commonly used for image stitching [7], but has also been successfully used in robotics computer vision [72].

The newly developed sensor and local map fusion technique can then be compared to algorithms relying on a available inter-sensor transformation [11].

5 Project Outline

This project is anticipated to be funded for three years by the *Studienstiftung des deutschen Volkes*, thus the plan is designed for a three year span.

The presented research is split up into different work packages:

- Literature study: In order to perform research based on state-of-the-art techniques, an extensive literature corpus has to be studied. Although this work package is a continuous task, most of the literature study should be completed before starting other tasks.
- Map Storage: Researching the usability of distributed hash tables as storage system and providing an implementation are the main tasks this work package. Additionally, the performance of various approaches can be evaluated.
- Distributed Path Planning: This work package consists of evaluating the usability of hierarchical path planning methods, performance comparison and an efficient implementation. While the underlying distributed map storage system does not necessarily have to be completed to start this task, it is recommended.
- Real-World Testing: The research of the previous work packets will be backed by simulation results. In order to learn the real-world capabilities of the presented approaches, testing on physical robots is required.
- Robustness Testing: This task includes additional research comparing the overall multi-agent mapping and planning approach to systems of single agents and other comparable setups.
- Thesis Writing: As this is a Ph.D. research project, a certain amount of time will be scheduled for writing a Ph.D. thesis.

The presented work packages are not entirely dependent on one another, but they are recommended to be scheduled one after another, in the order provided. The following table provides an estimate for the time required.

Objective	Start and end (months)
Literature study, identification of the main challenges to overcome	1-6
Development of a distributed, efficient, redundant and fault-tolerant map storage system	6-10
Researching distributed path planning using the aforementioned map storage	10-16
Implementation and real-world testing on available robots	16-22
Robustness testing with simulated and real world data	22 - 26
Ph.D. thesis writing	26-36

References

[1] Lars A.A. Andersson and Jonas Nygårds. On Multi-robot Map Fusion by Inter-robot Observations. In *12th International Conference on Information Fusion, 2009*, pages 1712–1721. IEEE, 2009.

[2] Rosario Aragues, Eduardo Montijano, and Carlos Sagüés. Consistent Data Association in Multi-robot Systems with Limited Communications. In *Robotics: Science and Systems*, pages 97–104, 2010.

[3] Tim Bailey and Hugh Durrant-Whyte. Simultaneous localization and mapping (SLAM): Part II. *IEEE Robotics & Automation Magazine*, 13(3):108–117, 2006.

[4] Maxim Batalin, Gaurav S. Sukhatme, and Myron Hattig. Mobile Robot Navigation using a Sensor Network. In *IEEE International Conference on Robotics and Automation, 2004. Proceedings.*, volume 1, pages 636–641. IEEE, 2004.

[5] Johann Borenstein and Liqiang Feng. Measurement and Correction of Systematic Odometry Errors in Mobile Robots. *IEEE Transactions on Robotics and Automation*, 12(6):869–880, 1996.

[6] Adi Botea, Martin Müller, and Jonathan Schaeffer. Near Optimal Hierarchical Path-Finding. *Journal of Game Development*, 1:7–28, 2004.

[7] Matthew Brown and David G. Lowe. Automatic Panoramic Image Stitching using Invariant Features. *International Journal of Computer Vision*, 74(1):59–73, 2007.

[8] Wolfram Burgard, Cyrill Stachniss, Giorgio Grisetti, Bastian Steder, Rainer Kummerle, Christian Dornhege, Michael Ruhnke, Alexander Kleiner, and Juan D Tardós. A Comparison of SLAM Algorithms Based on a Graph of Relations. In *IEEE/RSJ International Conference on Intelligent Robots and Systems, 2009*, pages 2089–2095. IEEE, 2009.

[9] Axel Bürkle, Florian Segor, and Matthias Kollmann. Towards Autonomous Micro UAV Swarms. *Journal of Intelligent & Robotic Systems*, 61(1-4):339–353, 2011.

[10] John Byers, Jeffrey Considine, and Michael Mitzenmacher. Simple Load Balancing for Sistributed Hash Tables. In *Peer-to-Peer Systems II*, pages 80–87. Springer, 2003.

[11] José A. Castellanos, José Neira, and Juan D. Tardós. Multisensor Fusion for Simultaneous Localization and Map Building. *IEEE Transactions on Robotics and Automation*, 17(6):908–914, 2001.

[12] Shi-Ming Chen, Jun-Feng Yuan, Fang Zhang, and Hua-Jing Fang. Multi-robot FastSLAM Algorithm Based on Landmark Consistency Correction. *Mathematical Problems in Engineering*, 2014, 2014.

[13] Yang Cheng, Andrew E. Johnson, Larry H. Matthies, and Clark F. Olson. Optical Landmark Detection for Spacecraft Navigation. *Advances in the Astronautical Sciences*, 114(3):1785–1803, 2003.

[14] Javier Civera, Andrew J. Davison, and J.M.M. Montiel. Dimensionless Monocular SLAM. In *Pattern Recognition and Image Analysis*, pages 412–419. Springer, 2007.

[15] Rik Claessens, Yannick Müller, and Benjamin Schnieders. Graph-based Simultaneous Localization and Mapping on the TurtleBot Platform. Retrieved from http://airesearch.de/written/Graph-Based-Simultaneous-Localization-and-Mapping-on-the-TurtleBot-Platform.pdf, 2013.

[16] John J. Craig. *Introduction to Robotics: Mechanics and Control*. Pearson, 3rd edition, 2013.

[17] Alexander Cunningham, Manohar Paluri, and Frank Dellaert. DDF-SAM: Fully Distributed SLAM using Constrained Factor Graphs. In *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 3025–3030. IEEE, 2010.

[18] Alexander Cunningham, Kai M. Wurm, Wolfram Burgard, and Frank Dellaert. Fully Distributed Scalable Smoothing and Mapping with Robust Multi-robot Data Association. In *2012 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1093–1100. IEEE, 2012.

[19] Edsger W. Dijkstra. A Note on Two Problems in Connexion with Graphs. *Numerische Mathematik*, 1(1):269–271, 1959.

[20] Albert Diosi and Lindsay Kleeman. Advanced Sonar and Laser Range Finder Fusion for Simultaneous Localization and Mapping. In *2004 IEEE/RSJ International Conference on Intelligent Robots and Systems, 2004. Proceedings.*, volume 2, pages 1854–1859. IEEE, 2004.

[21] Gamini M.W.M. Dissanayake, Paul Newman, Steve Clark, Hugh F. Durrant-Whyte, and Michael Csorba. A Solution to the Simultaneous Localization and Map Building (SLAM) Problem. *IEEE Transactions on Robotics and Automation*, 17(3):229–241, 2001.

[22] Esra Erdem, Doga Gizem Kisa, Umut Öztok, and Peter Schüller. A General Formal Framework for Pathfinding Problems with Multiple Agents. In *AAAI*, pages 290–296. AAAI Press, 2013.

[23] Erico Guizzo. How Google's Self-Driving Car Works. Retrieved from http://spectrum.ieee.org/automaton/robotics/artificial-intelligence/how-google- self-driving-car-works, 2011.

[24] John W. Fenwick, Paul M. Newman, and John J. Leonard. Cooperative Concurrent Mapping and Localization. In *IEEE International Conference on Robotics and Automation, 2002. Proceedings*, volume 2, pages 1810–1817. IEEE, 2002.

[25] Ettore Ferranti, Niki Trigoni, and Mark Levene. Brick & Mortar: An online multi-agent exploration algorithm. In *2007 IEEE International Conference on Robotics and Automation*, pages 761–767. IEEE, 2007.

[26] Martin A. Fischler and Robert C. Bolles. Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography. *Communications of the ACM*, 24(6):381–395, 1981.

[27] Franck Gérossier, Paul Checchin, Christophe Blanc, Roland Chapuis, and Laurent Trassoudaine. Trajectory-oriented EKF-SLAM using the Fourier-Mellin Transform applied to Microwave Radar Images. In *IEEE/RSJ International Conference on Intelligent Robots and Systems, 2009.*, pages 4925–4930. IEEE, 2009.

[28] Yi Guo and Lynne E. Parker. A Distributed and Optimal Motion Planning Approach for Multiple Mobile Robots. In *IEEE International Conference on Robotics and Automation, 2002. Proceedings.*, volume 3, pages 2612–2619. IEEE, 2002.

[29] Jungong Han, Ling Shao, Dong Xu, and Jamie Shotton. Enhanced Computer Vision with Microsoft Kinect Sensor: A Review. *IEEE Transactions on Cybernetics*, 43(5):1318–1334, 2013.

[30] Peter E. Hart, Nils J. Nilsson, and Bertram Raphael. A Formal Basis for the Heuristic Determination of Minimum Cost Paths. *IEEE Transactions on Systems Science and Cybernetics*, 4(2):100–107, 1968.

[31] Dominik Henrich. Fast motion planning by parallel processing – A review. *Journal of Intelligent and Robotic Systems*, 20(1):45–69, 1997.

[32] H.D. Herath, S. Kodagoda, and G. Dissanayake. Stereo Vision Based SLAM: Issues and Solutions. In *Vision Systems: Applications*, chapter 29, pages 565–582. InTECH, 2007.

[33] Johan D. Hose, Joseph Colineau, Cyrille Bichon, and Bernadette Dorizzi. Precise Localization of Landmarks on 3D Faces using Gabor Wavelets. In *First IEEE International Conference on Biometrics: Theory, Applications, and Systems, 2007*, pages 1–6. IEEE, 2007.

[34] Karl Iagnemma and Martin Buehler. Editorial for Journal of Field Robotics – special issue on the DARPA grand challenge. *Journal of Field Robotics*, 23(9):655–656, 2006.

[35] Been Kim, Michael Kaess, Luke Fletcher, John Leonard, Abraham Bachrach, Nicholas Roy, and Seth Teller. Multiple Relative Pose Graphs for Robust Cooperative Mapping. In *2010 IEEE International Conference on Robotics and Automation (ICRA)*, pages 3185–3192. IEEE, 2010.

[36] Hiroaki Kitano, Minoru Asada, Yasuo Kuniyoshi, Itsuki Noda, Eiichi Osawa, and Hitoshi Matsubara. RoboCup: A challenge problem for AI. *AI Magazine*, 18(1):73, 1997.

[37] Kurt Konolige, Dieter Fox, Benson Limketkai, Jonathan Ko, and Benjamin Stewart. Map Merging for Distributed Robot Navigation. In *IROS*, pages 212–217, 2003.

[38] Richard E Korf. Depth-first Iterative-Deepening: An Optimal Admissible Tree Search. *Artificial Intelligence*, 27(1):97–109, 1985.

[39] Jinyang Li, Jeremy Stribling, Thomer M. Gil, Robert Morris, and M. Frans Kaashoek. Comparing the performance of distributed hash tables under churn. In *Peer-to-Peer Systems III*, pages 87–99. Springer, 2005.

[40] Peter M. Ludwig. *Formation Control for Multi-Vehicle Robotic Minesweeping*. PhD thesis, Monterey, California. Naval Postgraduate School, 2000.

[41] Ryan Luna, Alexis Oyama, and K.E. Bekris. Network-Guided Multi-Robot Path Planning for Resource-Constrained Planetary Rovers. In *Proceedings of the 10th International Symposium on Artificial Intelligence, Robotics and Automation in Space (i-SAIRAS)*, pages 776–783. IEEE Computer Society Press, Los Alamitos, CA, 2010.

[42] Favio Masson, Jose Guivant, and Eduardo Nebot. Robust Navigation and Mapping Architecture for Large Environments. *Journal of Robotic Systems*, 20(10):621–634, 2003.

[43] Petar Maymounkov and David Mazieres. Kademlia: A Peer-to-Peer Information System Based on the XOR Metric. In *Peer-to-Peer Systems*, pages 53–65. Springer, 2002.

[44] Ian Millington and John Funge. *Artificial Intelligence for Games*. Taylor and Francis US, 2nd edition, 2009.

[45] Francesco Mondada, Michael Bonani, Xavier Raemy, James Pugh, Christopher Cianci, Adam Klaptocz, Stephane Magnenat, Jean-Christophe Zufferey, Dario Floreano, and Alcherio Martinoli. The e-puck, a Robot Designed for Education in Engineering. In *Proceedings of the 9th Conference on Autonomous Robot Systems and Competitions*, volume 1, pages 59–65. IPCB: Instituto Politécnico de Castelo Branco, 2009.

[46] Michael Montemerlo, Sebastian Thrun, Daphne Koller, Ben Wegbreit, et al. FastSLAM: A Factored Solution to the Simultaneous Localization and Mapping Problem. In *AAAI/IAAI*, pages 593–598. AAAI Press, 2002.

[47] Lauro Ojeda, Johann Borenstein, Gary Witus, and Robert Karlsen. Terrain Characterization and Classification with a Mobile Robot. *Journal of Field Robotics*, 23(2):103–122, 2006.

[48] Michael Otte and Nikolaus Correll. Any-Com Multi-robot Path-Planning: Maximizing Collaboration for Variable Bandwidth. In *Distributed Autonomous Robotic Systems*, pages 161–173. Springer, 2013.

[49] Wang Qiong, Yu Dengyun, and Jia Yang. Research Progress and Future Development on Mission Planning Technologies of Planetary Rover. In *2014 Asia-Pacific International Symposium on Aerospace Technology, APISAT2014*, volume 99, pages 1137 – 1145. Procedia Engineering, 2015.

[50] M. Rajesh, Roni Joseph, and T.S.B. Sudarshan. Fully Distributed and Decentralized Map Building for Multi-Robot Exploration. In *2014 International Conference on Embedded Systems (ICES)*, pages 220–224. IEEE, 2014.

[51] Andreagiovanni Reina, Luca Maria Gambardella, Marco Dorigo, and Gianni A. Di Caro. zePPeLIN: Distributed Path Planning Using an Overhead camera network. *International Journal of Advanced Robotic Systems*, 11:119, 2014.

[52] Michael Rubenstein, Christian Ahler, and Radhika Nagpal. Kilobot: A Low Cost Scalable Robot System for Collective Behaviors. In *2012 IEEE International Conference on Robotics and Automation (ICRA)*, pages 3293–3298. IEEE, 2012.

[53] Russell, Stuart and Norvig, Peter. *Artificial Intelligence: A Modern Approach*. Prentice-Hall, Second edition, 2002.

[54] Benjamin Schnieders. aiPath: Movement Planning in Physically Constrained Domains. Master's thesis, Maastricht University, July 2014. Retrieved from http://airesearch.de/written/aiPath-Movement-Planning-in-Physically-Constrained-Domains.pdf.

[55] G. Sharon, R. Stern, A. Felner, and N. Sturtevant. Conflict-Based Search For Optimal Multi-Agent Path Finding. In *AAAI Conference on Artificial Intelligence*, pages 563–569, 2012.

[56] David Silver. Cooperative Pathfinding. In *AIIDE*, pages 117–122. AAAI Press, 2005.

[57] Bruno Steux and Oussama El Hamzaoui. CoreSLAM: a SLAM Algorithm in less than 200 lines of C code. In *11th International Conference on Control, Automation, Robotics & Vision (ICARCV), 2010*, pages 1975–1979. IEEE, 2010.

[58] Ion Stoica, Robert Morris, David Karger, M. Frans Kaashoek, and Hari Balakrishnan. Chord: A scalable peer-to-peer lookup service for internet applications. *ACM SIGCOMM Computer Communication Review*, 31(4):149–160, 2001.

[59] Hauke Strasdat, J.M.M. Montiel, and Andrew J. Davison. Scale Drift-Aware Large Scale Monocular SLAM. In *Robotics: Science and Systems*, volume 2, page 5, 2010.

[60] Nathan Sturtevant and Michael Buro. Partial Pathfinding Using Map Abstraction and Refinement. In *AAAI*, volume 5, pages 1392–1397, 2005.

[61] Benjamin Suger, Gian Diego Tipaldi, Luciano Spinello, and Wolfram Burgard. An Approach to Solving Large-Scale SLAM Problems with a Small Memory Footprint. In *2014 IEEE International Conference on Robotics and Automation (ICRA)*, pages 3632–3637. IEEE, 2014.

[62] Marc Szymanski, Tobias Breitling, Jörg Seyfried, and Heinz Wörn. Distributed Shortest-Path Finding by a Micro-robot Swarm. In *Ant Colony Optimization and Swarm Intelligence*, pages 404–411. Springer, 2006.

[63] Duy-Nguyen Ta and Frank Dellaert. Linear-time Estimation with Tree Assumed Density Filtering and Low-Rank Approximation. In *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2014)*, pages 4556–4563. IEEE, 2014.

[64] Juan D Tardós, José Neira, Paul M Newman, and John J Leonard. Robust Mapping and Localization in Indoor Environments using Sonar Data. *The International Journal of Robotics Research*, 21(4):311–330, 2002.

[65] Sebastian Thrun, Michael Montemerlo, and Andrei Aron. Probabilistic Terrain Analysis For High-Speed Desert Driving. In *Robotics: Science and Systems*, pages 16–19, 2006.

[66] Ko-Hsin Cindy Wang and Adi Botea. M APP : a Scalable Multi-Agent Path Planning Algorithm with Tractability and Completeness Guarantees. *Journal of Artificial Intelligence Research*, pages 55–90, 2011.

[67] Craig Watman, David Austin, Nick Barnes, Gary Overett, and Simon Thompson. Fast Sum of Absolute Differences Visual Landmark Detector. In *ICRA'04. 2004 IEEE International Conference on Robotics and Automation*, volume 5, pages 4827–4832. IEEE, 2004.

[68] Changyun Wei, Koen V. Hindriks, and Catholijn M. Jonker. Multi-Robot Cooperative Pathfinding: A Decentralized Approach. In *Modern Advances in Applied Intelligence*, pages 21–31. Springer, 2014.

[69] Changyun Wei, Koen V. Hindriks, and Catholijn M. Jonker. Altruistic Coordination for Multi-Robot Cooperative Pathfinding. *Applied Intelligence*, pages 1–13, 2015.

[70] Stefan B. Williams, Gamini Dissanayake, and Hugh Durrant-Whyte. Towards Multi-Vehicle Simultaneous Localisation and Mapping. In *IEEE International Conference on Robotics and Automation, 2002. Proceedings.*, volume 3, pages 2743–2748. IEEE, 2002.

[71] Mark Woods, Andy Shaw, Iain Wallace, Mateusz Malinowski, and Philip Rendell. Demonstrating Autonomous Mars Rover Science Operations in the Atacama Desert. In *12th International Symposium on Artificial Intelligence, Robotics and Automation in Space i-SAIRAS*, 2014.

[72] Shao-Wen Yang and Chieh-Chih Wang. Multiple-model RANSAC for ego-motion estimation in highly dynamic environments. In *IEEE International Conference on Robotics and Automation, 2009.*, pages 3531–3538. IEEE, 2009.

[73] Kazuya Yoshida, Dragomir Nenchev, Genya Ishigami, and Yuichi Tsumaki. Space Robotics. In *The International Handbook of Space Technology*, pages 541–573. Springer, 2014.