RIGA TECHNICAL UNIVERSITY

ILZE ANDERSONE

THE DEVELOPMENT AND IMPLEMENTATION OF HYBRID MAP MERGING METHOD

SUMMARY OF DOCTORAL THESIS

Riga 2014

RIGA TECHNICAL UNIVERSITY

Faculty of Computer Science and Information Technology

Institute of Applied Computer Systems

ILZE ANDERSONE

THE DEVELOPMENT AND IMPLEMENTATION OF HYBRID MAP MERGING METHOD

SUMMARY OF DOCTORAL THESIS

Scientific advisor:

Dr.sc.ing., associated professor

A. ŅIKITENKO

Riga 2014

UDK 004.414.2(043)

An 138

Andersone I. The development and implementation of hybrid map merging method. Summary of Doctoral Thesis. -R.:RTU, 2014. - 58 p.

Printed in accordance with the decision of the board of Institute of Applied Computer Systems, Faculty of Computer Science and Information Technology, Riga Technical University No 87 of September 12, 2013.

This work has been supported by the European Social Fund within the project “Support for the implementation of doctoral studies at Riga Technical University”.

ISBN 978-9934-507-68-7

DOCTORAL THESIS
SUBMITTED FOR THE DOCTORAL DEGREE OF ENGINEERING SCIENCES
AT RIGA TECHNICAL UNIVERSITY

The defence of the thesis submitted for doctoral degree of engineering sciences will take place at an open session on September 12, 2014 in Meza Street 1/3, auditorium 202, Riga Technical University Faculty of Computer Science and Information Technology.

OFFICIAL OPPONENTS:

Professor, Dr.habil.sc.ing. Valerijs Zagurskis
Riga Technical University, Latvia

Professor, Dr.habil.sc.ing. Peteris Rivza
Latvia University of Agriculture, Latvia

Professor, PhD. Mart Tamre
Tallinn University of Technology, Estonia

APPROVAL

I confirm that I have developed this thesis submitted for the doctoral degree at Riga Technical University. This thesis has not been submitted for the doctoral degree in any other university.

Ilze Andersone ……………………………….. (signature)

Date: 04.04.2014.

The doctoral thesis is written in Latvian and includes introduction, 5 sections, conclusions, bibliography, 3 appendixes, 75 figures and 18 tables. The main text is 156 pages. The bibliography contains 91 references.

CONTENTS

INTRODUCTION **6**
1. MAPPING IN MULTI-ROBOT SYSTEMS **10**
1.1. THE MOTIVATION OF MULTI-ROBOT MAPPING 10
1.2. ROBOT MAP MERGING 11
1.3. MAP MERGING CHARACTERISTICS 11
1.4. THE SURVEY OF MAP MERGING METHODS 12
1.5. THE SUMMARY 14
2. MAP MERGING WITH UNKNOWN ROBOT POSITIONS **15**
2.1. LOCAL MAP MERGING 15
2.2. GLOBAL MAP MERGING 17
2.3. THE SUMMARY 18
3. REMMERG – METHOD FOR RELIABLE MAP MERGING **19**
3.1. MAP MERGING HYPOTHESIS AND THEIR REPRESENTATION 20
3.2. GLOBAL MAP MERGING 21
3.3. LOCAL MAP MERGING 26
3.4. THE SUMMARY 30
4. THE IMPLEMENTATION OF MAPPING SYSTEM **31**
4.1. THE DESCRIPTION OF THE MULTI-ROBOT SYSTEM 31
4.2. BINARY BAYES FILTER MAPPING 32
4.3. MODIFIED BINARY BAYES FILTER MAPPING 32
4.4. THE EVALUATION OF MAP MERGING SYSTEM 34
4.5. THE SUMMARY 35
5. THE IMPLEMENTATION OF MAP MERGING SYSTEM **36**
5.1. THE DESCRIPTION OF THE EXPERIMENTAL SOFTWARE SYSTEM 36
5.2. THE SELECTION OF MAP MERGING HYPOTHESIS EVALUATION PARAMETERS 36
5.3. EVALUATION OF REMMERG PERFORMANCE 38
5.4. SUMMARY 50
6. CONCLUSIONS AND RESULTS **51**
6.1. THE EVALUATION OF REMMERG METHOD 51
6.2. THEORETICAL RESULTS 53
6.3. PRACTICAL RESULTS 54
6.4. FURTHER RESEARCH 55
BIBLIOGRAPHY **56**

INTRODUCTION

Motivation of the research

The development of autonomous mobile robots is a research area of artificial intelligence since 1980s [Thr 2002]. The motivation of research is the many applications of the autonomous mobile robots that include precise agriculture, cleaning, grass mowing etc. One of the fundamental problems in mobile robotics that is still being solved is the problem of environment **mapping** [Thr 2005].

Mapping with multiple robots has several advantages over single robot mapping that makes it an attractive alternative: faster exploration, possibility of more accurate map creation, higher redundancy [Bur 2005, Thr 2002]. However, in multi-robot mapping also several new problems, specific to the use of multiple robots, arise. One of these problems is **map merging** – the merging of multiple robot's local maps into one common global map [Thr 2002].

The existing map merging approaches offer wide choice of different solutions, when the relative positions of the robots are known [Thr 1998, Sim 2000, Bur 2002, Ko 2003]. The map merging, when the relative positions are unknown, commonly considers the problem as the search for transformation between two maps (in this thesis called ***local map merging***) [Ami 2005, Bir 2006, Car 2008, Adl 2008], but, according to the results of literature analysis, there are no map merging methods that address the problem of reversible and reliable map merging in case of more than two robots (in this thesis called ***global map merging***).

The thesis considers the mapping and map merging in autonomous multi-robot system that takes into account both local and global map merging aspect. These aspects include the search for relative transformations of the maps and the general map merging process during the mapping with a special attention to the provision of map merging reversibility.

Goal of the thesis

The **goal of the thesis** is to develop and implement a map merging method for the acquisition of the multi-robot system global map, that implements reversible and dynamic map merging during mapping.

The approach developed in the thesis offers an alternative solution for the map merging and provides a way to deal with incorrect map merging cases. The method solves the map merging problem both in local (the merging of two maps without the information about their relative positioning) and global levels (the evaluation of the result, the reversibility and dynamism of the map merging). The developed method is implemented in the software system and its performance is evaluated with maps that are created by an autonomous robot system developed in Riga Technical University (by collaborating with Latvia University of Agriculture and SIA “Terra Virtuala”) during the project “The development of robotized intellectual multi-agent system technology” [RTU 2013].

Tasks of the thesis:

To achieve the goal, the following tasks have been specified:

- To analyse the situation in mapping with multi-robot systems and to identify the existing partial solutions in the problem domain of thesis – multi-robot mapping and map merging (chapter “Mapping in multi-robot systems”).
- To define the problem discussed in thesis and the requirements of the map merging method (chapter “The aspects of map merging”).
- To develop a method for multi-robot map merging that would deal with both local and global level map merging problem assuming that the relative positions of the robots are unknown (chapter “ReMMerg – method for reliable map merging”).
- To develop a mapping approach for the existing multi-robot system (chapter “The implementation of mapping system”).
- To implement the map merging method in software system and to experimentally evaluate its advantages and drawbacks (chapter “The implementation of map merging method”).
- To evaluate the performance and practical application of the developed method by practical experiments (chapter “Conclusions”).

Assumptions and restrictions

The developed map merging method is intended for use in multi- robot mapping systems in situations that hold the following assumptions:

- Every robot is able to independently create its local map that may be locally inaccurate but is globally accurate, i.e. small deviations of the actual environment configuration are acceptable but the overall map configuration is similar to the actual situation.
- Robots are able to communicate during mapping and to send their local maps to one central agent (robot or server).

Scientific novelty and practical value

The **scientific novelty** of the thesis is the development of a reliable map merging method that is based on the use of hypothesis tree data structure in map merging and the corresponding hypothesis tree manipulation algorithms. The method is based on the analysis of existing map merging approaches and the discovered inability of these approaches to reversibly and dynamically merge more than two local maps, when the relative positions of the robots are unknown.

The **practical value** of the thesis is the developed and implemented map merging method, that combines both local and global aspects of the map merging. The implementation of these aspects lets the method both to find mergings between two maps and to autonomously create the global map from all local maps with a possibility to reverse the map merging without losing local map updates after the merging. The **practical results** of the thesis are the following:

- The state of the art in the map merging field is summarized and analyzed.
- Based on the analysis of the map merging methods, two map merging aspects are identified and defined – local map merging and global map merging.
- A map merging method for a reliable map merging is developed and implemented.
- Map merging hypothesis evaluation approach is developed that allows to estimate the similarity of the common area of two maps by a known transformation. The approach takes into account the local inaccuracies of the maps.
- An algorithm for robot mapping is developed, that allows the map creation for robots with only close range sensors (such as impact sensors) available for mapping, if the robot position is known.
- It is experimentally demonstrated that the developed map merging method can autonomously produce a global map in a multi-robot system with more than two robots, identify conflicts in the proposed map merging hypotheses, exclude such hypothesis without data loss and create a new global map, taking into account previous experiences.

Approbation of the results:

1. Andersone I. The Characteristics of the Map Merging Methods: A Survey // Scientific Journal of Riga Technical University. Computer Sciences. - Applied Computer Systems. (2010) pp. 113.-121
2. Andersone I. Multi-Robot Map Merging in the Context of Image Processing // Scientific Journal of Riga Technical University. Computer Sciences. - 43. (2011) pp 124-130.

3. Andersone I. The Conceptual Model for reliable Multi-Robot Map Merging // Proceedings of Baltic Conference „Human-Computer Interaction”, Latvia, Riga, 23.-25. August, 2011.
4. Andersone I. The Influence of the Map Merging Order on the Resulting Global Map in Multi-Robot Mapping // Scientific Journal of Riga Technical University. Computer Sciences. - 44. (2012)
5. Andersone I., Liekna A., Nikitenko A. Mapping Implementation for Multi-Robot System with Glyph Localization // Scientific Journal of Riga Technical University, Computer Sciences, 2013
6. Andersone I., Liekna A. Robot Map Similarity Metric for Non-identical Maps // 12th International Scientific Conference on Engineering for Rural Development, Jelgava, May 2013
7. Andersone I., Nikitenko A. Reliable Multi Robot Map Merging for Inaccurate Maps // 12th Conference on Practical Applications of Agents and Multi-Agent Systems PAAMS'14 Salamanca, 4th-6th June, 2014

The structure of the thesis

The thesis consists of introduction, five chapters, conclusions, bibliography, glossary and acronym list.

Introduction describes the research motivation, defines the goal and tasks, and describes the scientific novelty and practical results of the thesis.

First chapter is dedicated to the theoretical survey of the multi-robot mapping, with the focus on map merging problem. It contains the multi-robot mapping motivation, map merging problem definition, map merging characteristics identification and the analysis of the existing map merging methods. Second chapter focuses on the map merging when the relative positions of the robots are unknown. This chapter defines local and global map merging aspects and analyses the map merging state-of-the-art in each of these aspects. It is demonstrated that the map merging sequence can significantly influence the acquisition of the global map, and that there is a necessity for a map merging method which allows to constitute the global map in different ways based on the previous experience. Third chapter describes the developed map merging method that implements both map merging aspects – local map merging and global map merging. Fourth chapter describes the developed mapping system and the proposed mapping approach for robots with close range sensors. Fifth chapter is dedicated to the implementation of the developed map merging method and the experimental results. The thesis ends with conclusions.

1. MAPPING IN MULTI-ROBOT SYSTEMS

The development of **autonomous mobile robots** is a popular research area of artificial intelligence since 1980ies [Thr 2002]. The motivation of research is the numerous applications of autonomous mobile robots that include planet exploration, scouting, rescue missions, cleaning, mowing etc. [Bur 2002]. One of the fundamental problems in mobile robotics is **the mapping** of the environment [Thr 2005]. The map of the environment that could be used by robots is not always readily available. The building drawings often do not represent the actual situation, and even precise drawings do not contain furniture and other objects that impact the robot movement. The robot's ability to independently create the map of environment significantly reduces the difficulty of introducing the robot to new places and allows the robot to adapt to the changes more successfully [Thr 2005].

The **mapping with one robot** is an extensive research area and there are still some unsolved problems. The problems most often mentioned in the literature are [Thr 2002]:

- Measurement errors,
- Simultaneous Localization and Mapping (SLAM),
- Data correspondence problem,
- Robot navigation,
- Mapping multi-dimensionality,
- Dynamic environment.

1.1. The motivation of multi-robot mapping

Some of the problems mentioned above can be easier solved by using **multi-robot mapping –** for example, measurement errors and data correspondence [And 2009]. Despite that the multi-robot mapping in general is more complicated than single robot mapping. However, there are several advantages that make the multi-robot mapping an attractive alternative to single robot mapping [Bur 2005, Thr 2002]:

- Multiple robots can explore the environment faster that a single robot.
- Multiple robots ensure the duplication of system functions (redundancy) and the system becomes more fault-tolerant.
- Multiple robots can create more accurate maps, if they are able to recognise each other and determine their relative positions [Fox 2000, Mar 2005].

1.2. Robot map merging

The origin of the robot team concept are the late 1980ties [Par 2000], but only in the last twenty years an intensive research in multi-robot mapping has been performed. Although the robot teams offer several advantages over single robot platforms, several new problems arise that are specific to multi-robot mapping case [And 2009].

Out of all the multi-robot mapping problems this thesis considers the map merging problem, which is a relatively new research area. When the robot explores the environment, it collects the information with its sensors. If the environment is being explored with several robots, the information they collect must be used to create one common global map. The **map merging** is the merging of map information from several robots into one global map [Ko 2003].

1.3. Map merging characteristics

To determine the actual tendencies and problems in multi-robot map merging, the existing map merging methods were analyzed [Ish 1993, Thr 1998, Sim 2000, Roy 2000, Ded 2000, Thr 2001, Bur 2002, Rou 2002, Wil 2002, Ko 2003, Kon 2003, Thr 2002, Rod 2004, Hua 2005, Lak 2005, Ho 2005, Ami 2005, Car 2005, Bir 2006, Adl 2008, Car 2008, Guo 2008, Aln 2010, Bal 2010, Top 2010]. In the result of analysis several characteristics were identified that must be taken into account to solve the map merging problem in multi-robot system:

- **Relative coordinate systems of robots**. All map merging methods can be divided in three groups depending on the moment when the relative coordinate systems are acquired and maps are merged: 1) The relative positioning is known from the beginning; 2) Initial relative positions of robots are unknown but they are acquired during mapping when robots meet; 3) Initial relative positions of robots are unknown and are never acquired during mapping.
- **Map type**. The maps created by robots can be very diverse: in the literature most commonly used maps are metric occupancy grids [Bir 2006, Top 2010, Guo 2008]. Some researchers use different metric maps [Adl 2008, Aln 2010] or topological maps [Hua 2005].
- **Information used in map merging.** The map merging is performed 1) by using initially known or during mapping determined relative positioning of the robots or 2) by proposing a hypothesis about the relative positioning of the maps by

searching the map transformation space – all possible rotations and translations of one map against the other map.

- **Map merging time**. The time necessary describes whether the map merging can be performed during mapping without significantly delaying the robotic system performance.
- **Map accuracy**. The maps can be accurate, locally inaccurate and globally inaccurate.

1.4. The survey of map merging methods

Several map merging method classifications can be found in literature. The most common classification is by the relative positioning or coordinate systems of the robots [Ko 2003, Hua 2005, Ami 2005, Ho 2005, Bir 2006] and by the map type [Hua 2005, Bir 2006]. In this thesis the relative coordinate positioning classification is used.

Initially multi-robot mapping approaches were simply extended one robot mapping approaches (the existing methods were adjusted for use in multi-robot systems instead of developing new methods specially designed for multi-robot teams) [Par 2000]. The map merging problem was similarly simplified by assuming that the robots create their maps in common reference frame. To use this approach the robots must know their initial relative positions. Several authors have developed map merging methods based on this assumption [Thr 2001, How 2006, Cec 2006]. In the case of known relative positions the robots merge their maps during the mapping by incorporating their sensor data into one common global map.

Different algorithms can be used for robotic mapping, however, most of mapping methods, when robot relative positions are known, use one of four approaches [Che 2007]:

- Expectation maximization methods – the map with the highest probability is created that is based on robot sensor data sequences [Thr 2002].

- Kalman Filter methods – widely used signal processing methods that are also successfully used in robotics in mapping, localization and other fields [Thr 2006]. Sensor data is used to create a map that contains the aposterior probabilities about the locations of environment features [Thr 2002].

- Particle filter methods that represent the possible robot and object locations in the map as particle sets.

- Set membership methods. In this method group the locations of robots and environment features are defined as areas where, based on the available information, they are located for sure.

All the multi-robot mapping approaches mentioned above assume that the relative positions of the robots are known initially or are acquired during the mapping. If the positioning information is unavailable, these approaches are unable to merge maps. The methods that assume unknown relative positions of the robots are addressing the more complicated case of map merging and use transformation approach to solve it – they rotate and translate one map against the other in attempt to find the best possible merging.

The map merging without known relative positions differ in complexity depending on the type of maps. Metric and topological maps can be created during mapping, and generally topological map merging is simpler than metric map merging [Hua 2005].

The merging of topological maps is simpler than metric map merging because there is additional structured information available in the search for the common parts – the graphs [Hua 2005]. Several researchers have addressed the problem of topological map merging and offered solutions usable in practice [Hua 2005, Ded 2000].

The merging of metric maps is more complex than topological map merging because it is not possible to use graph comparison approaches. To simplify the problem of metric map merging, many researchers use metric maps that are supplemented with additional information [Kon 2003, Adl 2008, Ami 2005, Lak 2005, Ho 2005].

Only few researchers have addressed map merging problem, when only geometric information is available. One of the first map merging methods of this type was developed by Carpin and Birk [Car 2005]. The method uses adaptive random walk algorithm that rotates and translates occupancy grid maps to find the best transformation. The transformation is evaluated by using image similarity metric. In [Bir 2006] this map merging approach is supplemented with map similarity evaluation that determines the reliability of the map merging result. The best transformation is the transformation with the lowest image similarity metric function value that is computed in every step of the search algorithm [Bir 2005].

Another map merging approach that was developed by Carpin uses Hough Transformation [Car 2008]. This method in its current development stage requires local maps that contain straight lines. The main idea of the method [Car 2008] is the acquisition of Hough spectres from the robots' local maps and the search of correlations between two spectres. The Hough spectres represent the most common directions of the straight lines in the maps. The maximums of the correlations represent the possible rotations of map transformation hypothesis. To find the translations on X and Y axis, two additional spectres are computed – X and Y spectres.

1.5. The summary

Based on the survey of existing map merging methods, most methods consider the case, when the positions of the robots are known initially or are acquired during the mapping, even though this condition implies that the robots must be specifically placed at the start of mapping or must be able to recognise each other and evaluate their relative positions.

In the case of unknown relative positions of the robots, the map merging hypothesis is acquired by using heuristics. The best heuristic evaluation does not guarantee the correct result, and it is possible that the hypothesis with the highest heuristic evaluation is not found. Therefore, when compared with map merging that uses the positional information of the robots, map merging with heuristics is much more likely to propose an incorrect map merging hypothesis.

It can be concluded that the map merging decision must be reversible, and it especially important, when the relative positions of the robots are unknown. Despite that, virtually all map merging methods consider the map merging without positional information as a search for transformation between two local maps.

2. MAP MERGING WITH UNKNOWN ROBOT POSITIONS

The map merging decision must be easily reversible, especially in case of unknown robot relative positions. Therefore map merging must take into account two map merging aspects:

- **Local map merging**. In this aspect map merging is **merging of two robots' local maps** – the search for common part in both maps, the merging of two maps by using the discovered transformation and the evaluation of the result.
- **Global map merging.** In this aspect the map merging is **map merging in the context of mapping.** Additionally to the local aspect of merging, global map merging considers, how the maps for merging are chosen. It allows to reject an existing map merging hypothesis during mapping and to merge maps several times, based on the previous experience.

2.1. Local map merging

In the context of this thesis the local map merging is **the merging of two robots' local maps -** the search for common part in both maps, the merging of two maps by using the discovered transformation and the evaluation of the result.

General local map merging process can be seen in Figure 2.1. The map merging methods consist of three main steps:

- **Feature identification**. The acquisition of the features specific to the maps.
- **Search**. The search of relative transformation between the maps that is based on the identified features. The feature identification and the search in the context of one method are often tightly connected.
- **Map similarity evaluation**. The map similarity evaluation is an independent step that can be adapted for any map merging method.

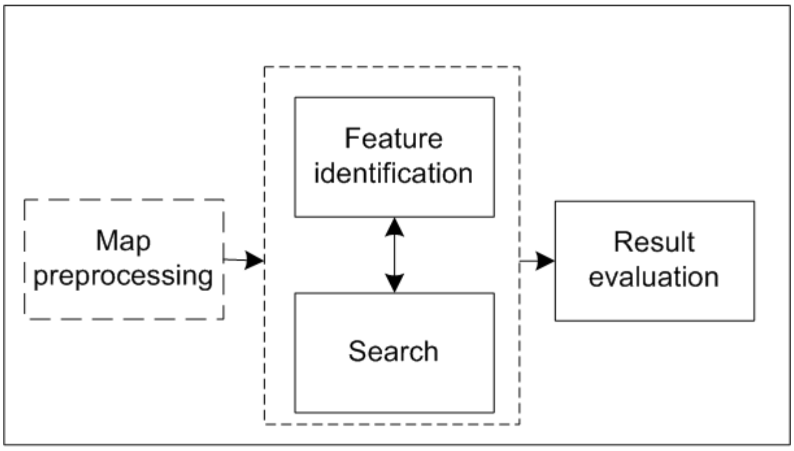


Figure 2.1. General map merging process

2.1.1. Features and search strategies

The search strategies used in map merging are based on the acquired features. For example, in [Car 2008] the search strategy is based on the Hough spectres acquired from maps. The correlations between these spectres show possible rotations. [Lak 2005] search strategy is based on the comparison of specific lines.

The different search strategies used for different features means that the feature identification and search strategy of different methods are usually incompatible. However, there are cases when it is possible. For example, different search algorithms can be used together with map similarity metric used in [Bir 2006].

2.1.2. The evaluation of map merging hypothesis

To evaluate the proposed map merging hypothesis, a numerical evaluation of map merging hypothesis must be introduced. Although the introduction of this evaluation does not guarantee correct map merging result, it helps to discard obviously incorrect transformations. The rejection of the map merging hypotheses must be automatic without the involvement of humans. To achieve this, two numerical values are required [Bir 2006]:

- **The evaluation of the map merging hypothesis** – an evaluation that describes the similarity of the common area of two maps by the current transformation hypothesis.
- **The map merging hypothesis confirmation threshold** – if the evaluation of the map merging hypothesis exceeds this threshold, the two maps are considered acceptably merged and the map merging hypothesis is confirmed.

2.2. Global map merging

Most of the research of the map merging by unknown relative positions considers only local map merging [Bir 2006, Car 2008, Lak 2005, Top 2010]. Only few researchers have considered the global map merging problem [Kon 2003, Hua 2005], but none has offered a real solution to this problem.

Global map merging is a problem of acquiring the global map during multi-robot mapping without losing information even if incorrect mergings have been made. To achieve this goal, the map merging method must be capable of creating different map merging sequences as demonstrated by the map merging experiment further in this chapter.

The map merging sequence can significantly influence the global map and wrong map merging hypotheses may be proposed. Three maps used to demonstrate this importance can be seen in the figure 2.2. All the maps have a common part.

Figure 2.2. Maps used for experiments map$_1$, map$_2$ un map$_3$.

Figure 2.3 shows two results of the map merging that are acquired by merging the same three maps in different sequence. Both resulting maps were acquired by merging the maps in Figure 2.2 and in each merging choosing the transformations with the highest hypothesis evaluation value.

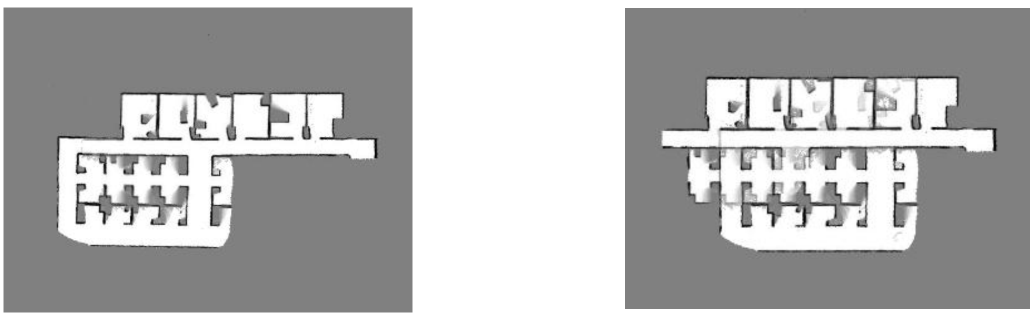


Figure 2.3. The maps acquired from merging three maps in different sequence. Left: successful map merging (map$_1$ and map$_{2+3}$) Right: unsuccessful map merging (map$_{1+2}$ and map$_3$)

Figure 2.3 shows two results of the map merging that are acquired by merging the same three maps in different sequence. The different results show that the map merging method must be able to choose different map merging sequence in case the current sequence does not return an acceptable result. Also the evaluation of map merging hypothesis must be able to reject obviously inconsistent hypotheses.

2.3. The summary

In this chapter the author of thesis arguments that the map merging decision must be easily reversible and it is especially important, when the relative positions of the robots are unknown. Therefore two aspects of map merging must be taken into account: local map merging and global map merging. Different methods exist to perform local map merging for different types of maps but the global map merging by unknown positions is virtually untouched in the map merging literature.

The map merging sequence significantly influences the global map and it is possible that incorrect map merging hypotheses are proposed. It proves the point that the global map merging aspect must be implemented in map merging method, especially when the relative coordinate frames of the maps are not known.

Without safe information about the map overlaps it is only possible to propose a map merging hypothesis and to test the plausibility of this hypothesis during further mapping. If the hypothesis turns out to be incorrect, there must be a way to preserve the original maps of the robots without the loss of information.

3. REMMERG – METHOD FOR RELIABLE MAP MERGING

This chapter describes the map merging method for reliable map merging offered by the author – ReMMerg (from **Re**liable **M**ap **Merg**ing method).

The probability of merging two robot local maps correctly is never ‘1.0’, and there is always at least small possibility of error. It means that one of the tasks of map merging method is to ensure the possibility to merge maps **reversibly**. The reversible map merging is especially important, if the relative positions of the robots are not available, because the common area of map is unknown. In this case it is only possible to propose a map merging hypothesis and to verify the hypothesis during further mapping. To avoid the loss of information acquired after the merging in case of hypothesis rejection, there must be a way to store the local maps of the robots.

Another important task of map merging is the decision making about the moment, when it can be considered that the proposed map merging hypothesis is believable [Kon 2003]. In this thesis **believable map merging hypothesis** is a hypothesis, whose map merging hypothesis evaluation exceeds an empirically set threshold. The **map merging hypothesis evaluation** is a number in interval [0; 1] that characterises the similarity of the common part of two maps by the proposed map merging hypothesis.

In this thesis the **reliable map merging** is a map merging that fulfils three requirements compliant to the arguments above:

- It provides the reversibility of map merging – at any mapping point it is possible to return to the point before the map merging without losing information acquired by individual robots after the merging.
- It provides map merging dynamism – the ability to offer different map merging sequences for the acquisition of the global map by taking into account the previous mapping experience.
- The decision to merge two local maps is only made, if the evaluation of the proposed map merging hypothesis exceeds previously empirically set threshold (if the map merging hypothesis is believable).

A general structure of ReMMerg method is depicted in Figure 3.1.

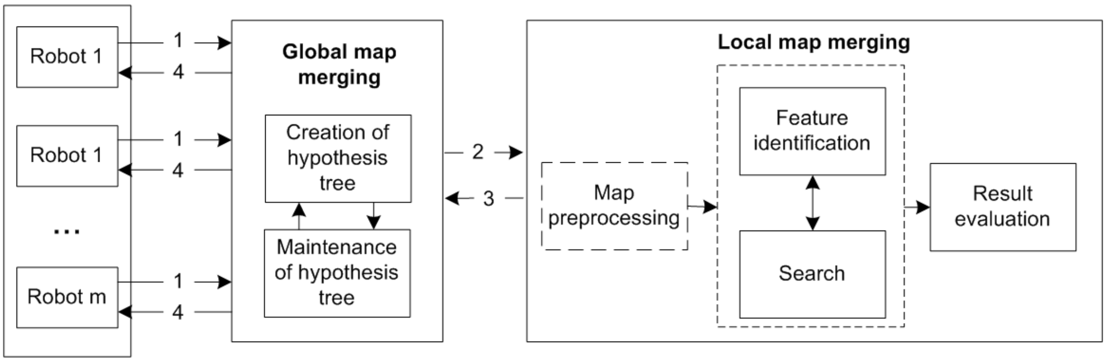


Figure 3.1. The general structure of the ReMMerg method

Information flows in Figure 3.1.: 1) robot local maps; 2) map pairs and the rejected hypothesis list; 3) map merging hypotheses and messages about the success/failure of map merging; 4) hypothesis tree and all local maps of the robots

ReMMerg method has two main parts, each of which implements one map merging aspect: Global map merging and Local map merging. Both parts are important for the ensuring of map merging reliability and reversibility. Local map merging part is responsible for search of map transformations, map combining and the evaluation of the result. Global map merging part implements the functions of hypothesis tree creation and hypothesis maintenance.

3.1. Map merging hypothesis and their representation

To provide the reversibility of map merging, it is important to choose an appropriate structure for storing local and global maps. Otherwise, as argued in [Kon 2003], if robots merge their local maps and continue map merging with a common global map, it can be complicated to separate maps without losing information that is acquired after the merging.

Map merging cannot be an irreversible action and from it one can conclude that the map merging result is not a global map but a map merging hypothesis. **The map merging hypothesis** is a triple, and its elements are two maps or hypotheses and their mutual transformation (equation 3.1).

`<map ∨ hypothesis, map ∨ hypothesis, transformation>` (3.1.)

The format of hypothesis is graphically depicted in Figure 3.2. The hypothesis contains information about the two merged maps and the transformation between them. The transformation is the positioning of the second map relative to the first map – the translations on X and Y axis and the rotation.

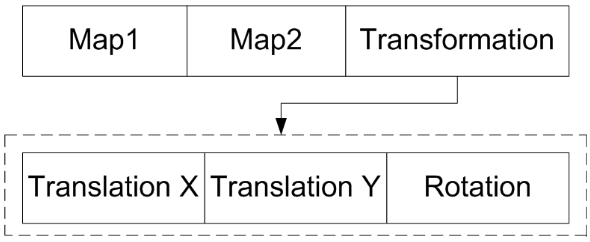


Figure 3.2. The representation of map merging hypothesis

Such representation of hypothesis allows to use the local maps in the creation of global map and to reject hypotheses at any time without the need to restore the maps of all proposed hypotheses.

3.2. Global map merging

The ReMMerg global map merging part is responsible for the maintenance of hypothesis tree and the creation of the global map. The map pairs to be merged are selected based on the previous mergings. The Figure 3.3. shows the process of global map merging process.

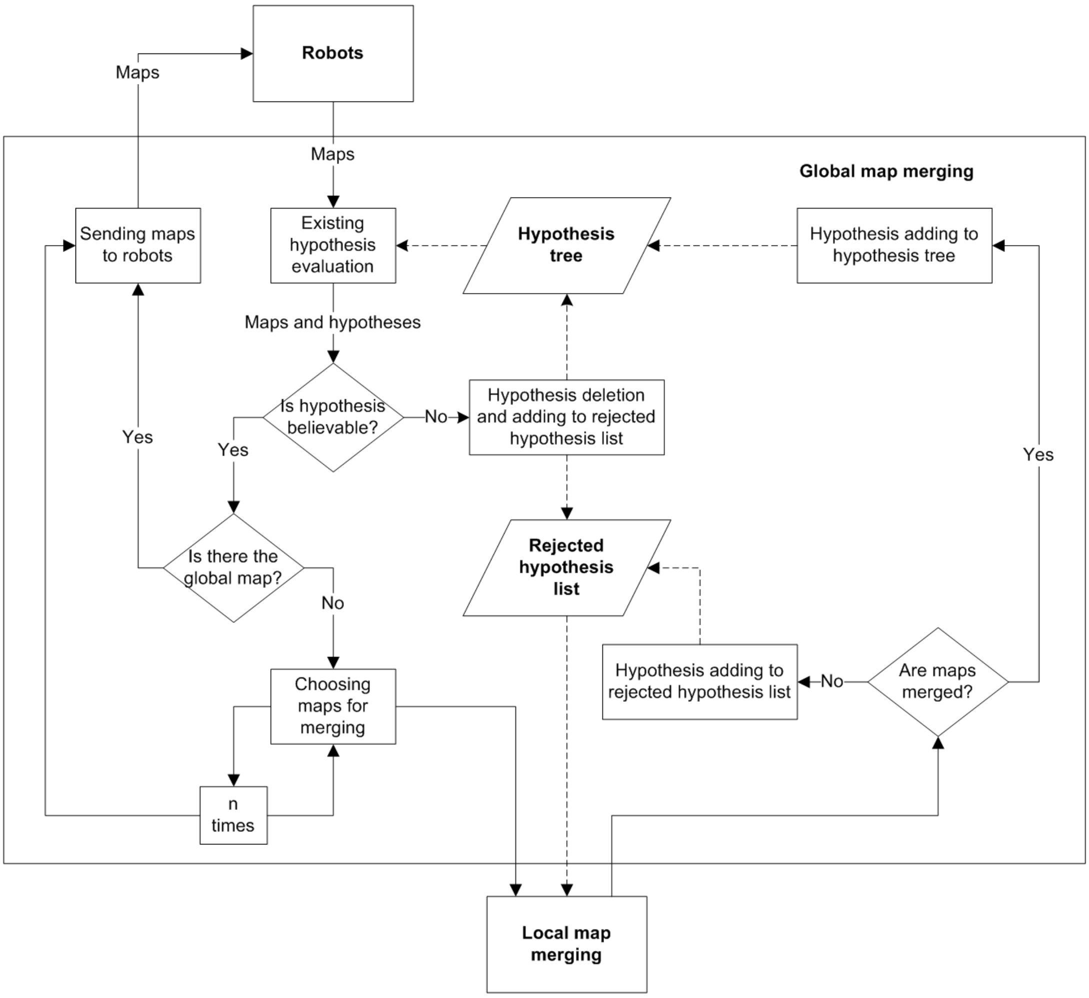


Figure 3.3. Global map merging process

The global map merging process that is depicted in Figure 3.3. is the following:

- Robot receives one or more local maps and hypothesis trees from other robots.
- All hypotheses in the local hypothesis tree are evaluated. If a hypothesis is discovered to be incorrect, then this hypothesis and all the dependent hypotheses are deleted from hypothesis tree, and the rejected hypothesis list is updated.
- If the hypothesis tree contains several highest level maps or the highest level map does not contain all local maps, then the local map merging is performed.
- In the case of successful local map merging the local hypothesis tree is updated.
- In the case of unsuccessful local map merging the proposed hypothesis is added to the rejected hypothesis list.

3.2.1. Data structures used for map merging

ReMMerg method uses three specific data structures, each of which performs and important role in map merging:

1. Hypothesis tree

One map merging hypothesis is not sufficient, if the environment is being explored by more than two robots. In literature the author has not encountered any map merging method that would be able to merge more than two maps simultaneously without the positional information. Therefore the global map is created gradually by merging local maps sequentially. In the ideal case the global map can be acquired by performing *n-1* map mergings (where *n* is the count of local maps). In such case each map is used exactly once for the merging.

In this thesis a structure is defined that represents the dependencies between map merging hypotheses - the hypothesis tree. Formally **the hypothesis tree** is a set of full binary trees, that corresponds to the following conditions:

- The leaf nodes of the tree are the local maps, and these nodes are unique in the whole tree set, i.e., every local map is represented as a leaf node in the tree set exactly once.
- Every tree node, that is not a leaf node, represents one map merging hypothesis, and the children of this node are local maps and/or hypotheses, the merging of which is the basis of hypothesis.
- Every tree root is the highest level map merging hypothesis.

The **highest level map merging hypothesis** is a hypothesis that is not involved in the creation of any other hypothesis, or its node is not a child of any other node. If the hypothesis tree set has only one binary tree, then its root or highest level map merging hypothesis is the **global map hypothesis**. Figure 3.4. shows examples of hypothesis trees that contain one and two trees.

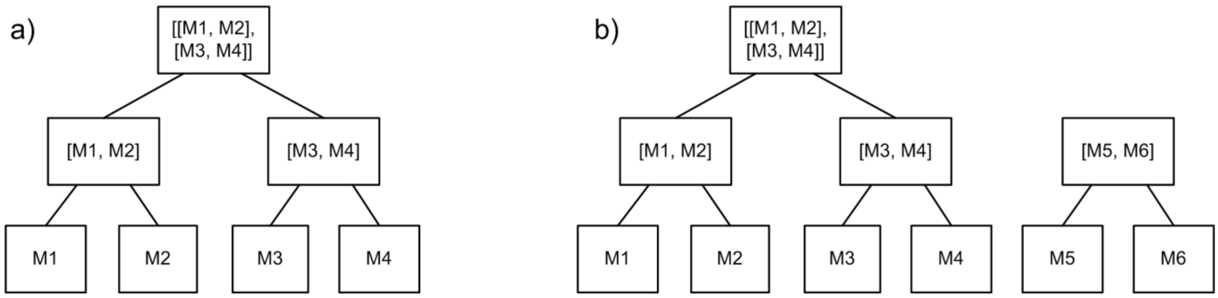


Figure 3.4. Examples of hypothesis tree: a) one tree set, b) two tree set

2. Map merging history

ReMMerg method used map merging history to ensure diversity in map merging attempts. The **map merging history** is a list, where each element of the list is a triple (equation 3.2.): first map, second map and the count of merging attempts for the map pair. The sequence of the maps is not important, and every map pair appears in the list only once. .

`<map1, map2, merging count of the map pair>` (3.2.)

3. Rejected hypothesis list

To avoid the proposal of hypothesis that were previously recognized as incorrect, the ReMMerg method maintains the rejected hypothesis list. The **rejected hypothesis list** is a list, that stores hypothesis, whose evaluation is lower than map merging hypothesis confirmation threshold.

3.2.2. Hypothesis tree evaluation and hypothesis deletion

The local maps created by robots are modified during mapping, and hypotheses that are believable in the earlier mapping stages, may turn out to be incorrect later. To avoid the proposal of new hypotheses on the base of incorrect hypotheses, in ReMMerg the existing hypotheses are evaluated in each iteration. The same evaluation algorithm can be used to evaluate these hypothesis as in the local map merging (a detailed description of this algorithm can be found in chapter 3.3.2).

If the evaluation of the map merging hypothesis is lower than the set hypothesis confirmation threshold, then the hypothesis is rejected and added to the rejected hypothesis list. The hypothesis deletion can be performed by following one of several scenarios.

1. The deletion of the highest level hypothesis

The highest level hypothesis is a hypothesis, on the basis of which no other hypothesis are proposed. In the scenario of deleting the highest level hypothesis it is deleted and added to the rejected hypothesis list. In figure 3.5. (and further in figures 3.6. and 3.7.) the bold line shows the deleted hypothesis, and the gray colour shows the hypothesis in the rejected hypothesis list.

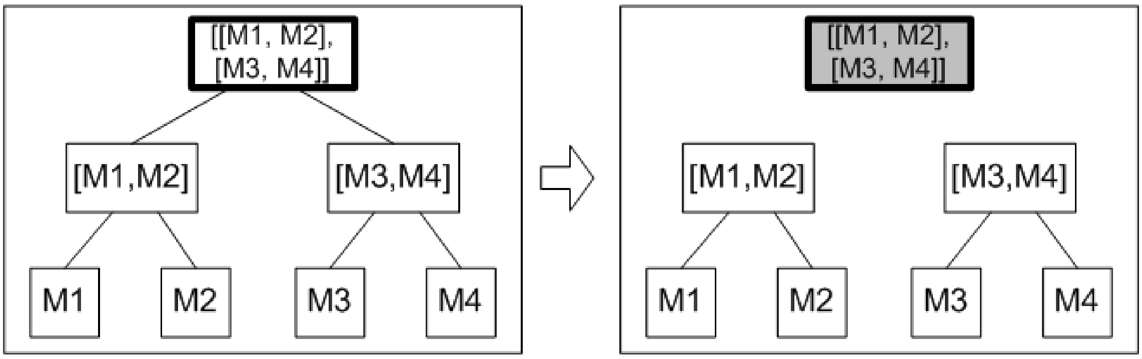


Figure 3.5. An example of deleting the highest level hypothesis

2. The deletion of hypothesis with dependencies in hypothesis tree

If the hypothesis that does not belong to the highest level is rejected and deleted, for example, H1 = <M1, M2>, and it is a part of at least on other hypothesis (for example, H2 = <<M1, M2>, <M3, M4>>), then all the dependent hypothesis are also deleted but only lowest level hypothesis is added to the rejected hypothesis list. An example in Figure 3.6. shows that only one hypothesis H1 is added to the rejected hypothesis list, because the proposal of H2 is no longer possible.

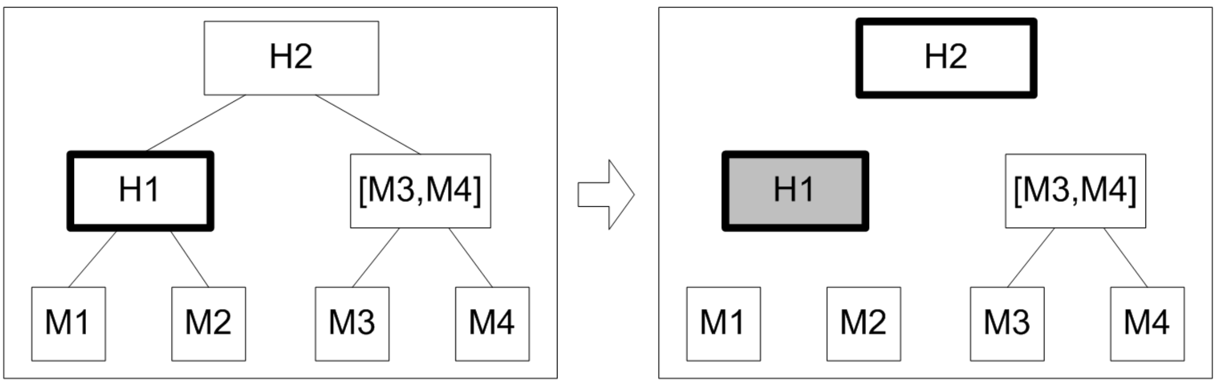


Figure 3.6. An example of deleting hypothesis with dependencies in hypothesis tree

3. The deletion of hypothesis with dependencies in rejected hypothesis list

If a hypothesis is deleted that has dependent hypotheses in rejected hypothesis list, then the rejected hypothesis is added to the rejected hypothesis list but all the dependent hypotheses are discarded from this list. The dependent hypotheses can't be proposed again because it is no longer possible to propose the deleted hypothesis.

3.2.3. The selection of maps to be merged

The problem of choosing map merging count is solved by computing the highest level hypothesis count in the current hypothesis tree. If the highest level has *n* hypotheses, then the minimal merging count for the acquisition of the global map is *n – 1*, and the maximal count is C_n^2 (by checking all the possible highest level hypothesis combinations). Equation 3.3. shows the possible map merging count interval:

$$s \in [n - 1;\ \mathrm{C}_\mathrm{n}^2] \tag{3.3.}$$

s – the necessary map merging count, n – the count of highest level hypotheses in the hypothesis tree, C_n^2 – maximum possible combinations from n by 2.

In a particular application the map merging count can be chosen based on the previously known information about the robot system. For example, if the average time of local map merging and maximum available time for map merging is known, then the merging count can be acquired by using equations 3.4. and 3.5.:

$$s = t_{max} / t_{lok} \tag{3.4.}$$

$$s = \begin{cases} n - 1, if\ s < n - 1 \\ \mathrm{C}_\mathrm{n}^2, if\ s > \mathrm{C}_\mathrm{n}^2 \\ s, if\ s \in [n - 1;\ \mathrm{C}_\mathrm{n}^2] \end{cases} \tag{3.5.}$$

s – the necessary map merging count, t_{max} – time available for map merging, t_{lok} – the average time of local map merging, n – the count of highest level hypotheses in the hypothesis tree, C_n^2 – maximum possible combinations from n by 2.

When the count of map merging attempts is known, the map pairs for the merging must be chosen. Both local maps of the robots and map merging hypothesis are eligible for the merging. The selection of map merging pairs is sequential, and each pair is only chosen, when previous map merging attempt is finished. This way the results of the previous mergings are taken into account in the selection of the next merging, and that allows selecting recently proposed map merging hypothesis for the next merging.

For selection of map pairs the map merging history is used, that maintains the count of each possible map pair mergings. Map merging history contains all the possible combinations of local maps and hypotheses.

3.2.4. The maintenance of hypothesis tree

The hypothesis tree is updated in two cases:

1. The adding of hypothesis

A new hypothesis is added to the hypothesis tree in case, when a local map merging attempt is performed, and the evaluation value of the hypothesis is equal or higher than the set hypothesis confirmation threshold.

2. The deletion of hypothesis

The hypothesis is deleted from the hypothesis tree, if it is discovered during the hypothesis tree inspection that the hypothesis is no more believable – that is, its evaluation no longer exceeds the hypothesis confirmation threshold. If the hypothesis is deleted, then it is added to the rejected hypothesis list, so that it is not proposed repeatedly.

3.3. Local map merging

The local map merging part in ReMMerg method implements the aspect of local map merging – the search for transformation between two maps, map merging by using found transformation and the evaluation of the result. As a result of local map merging **the map merging hypothesis** is proposed.

The local map merging process in ReMMerg is depicted in Figure 3.7. The map pair to be merged and the rejected hypothesis list is received from the global map merging part. The result of local map merging is either map merging hypothesis or a message about the failure of map merging attempt.

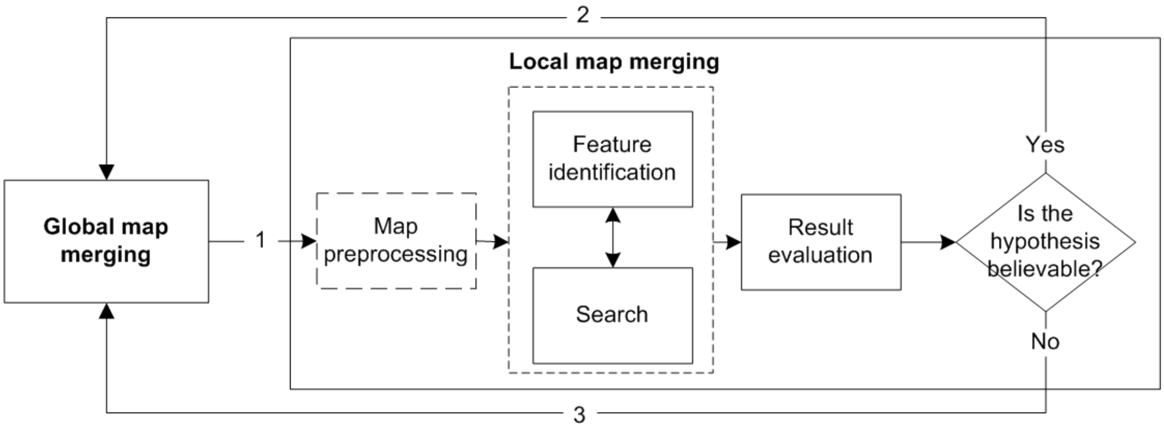


Figure 3.7. The local map merging process

Information flows in figure 3.7.: 1) the map pair to be merged and the rejected hypothesis list; 2) map merging hypothesis; 3) message about the failure of map merging.

3.3.1. Local map merging by using Hough Transformation

From the local map merging methods considered in chapter 1, the map merging by using Hough Transformation was chosen for use in this thesis [Car 2008]. This method was chosen because it allows the merging of occupancy grid maps during mapping, when the relative positions are unknown. Additionally, as demonstrated further in the chapter 5, this local map merging method is capable of merging locally inaccurate maps and can propose multiple map merging hypotheses [Car 2008].

The main idea of the local map merging by using Hough Transformation is the acquisition of Hough spectres from the local maps and the search for correlations between two spectres [Car 2008].

3.3.2. EvaLIM – Hypothesis evaluation for locally inaccurate maps

The evaluation of the map merging hypothesis must be tolerant to the local inaccuracies of the maps for the successful use in real life systems. Local inaccuracies in maps represent the robot sensor measurement errors and the local deviations of position estimate from the actual robot location. The imperfection of the sensors and effectors does not allow the creation of perfectly accurate maps [Thr 2006], therefore all the maps created in real robotic systems are locally or globally inaccurate. The author of thesis has not encountered any evaluation methods that are capable of evaluating map merging hypotheses for locally inaccurate occupancy grid maps.

In the thesis a map merging hypothesis evaluation EvaLIM (short for **Eva**luation of Map Merging Hypothesis for **L**ocally **I**naccurate **M**aps) is developed that is capable of evaluating the similarity of locally inaccurate maps, and allows to change the influence of particular cell types on the result. The EvaLIM is computed by using equation 3.6.:

$$SM_{m_1,m_2} = w_{occ} * s_{occ} + (1 - w_{occ}) * s_{free} \tag{3.6.}$$

w_{occ} – 'occupied' cell weight, s_{occ} – 'occupied' cell similarity evaluation, s_{free} – 'free cell similarity evaluation

The weight of 'occupied' cells can assume any value from 0 to 1. The more this value exceeds 0.5, the more influence 'occupied' cell similarity has on the result, and otherwise.

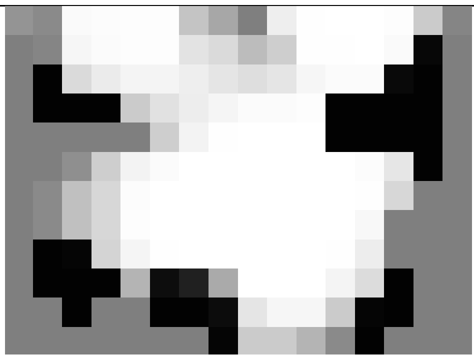
To compute the similarity of cells one parameter is required – the distance threshold d_{max} that describes, how far the mapping error can extend in the particular mapping system. This threshold defines the Manhattan distance, in which two cells are considered to be 'within reach of each other' – sufficiently close to possibly represent the same obstacle.

The cell similarity is computed by creating and using the **distance grids**. The distance grid of the map represents the Manhattan distance of each cell to the closest cell with a predefined target value (in this case it is a cell with a value 'occupied' or 'free') [Bir 2006].

To adapt the distance grids for the specifics of robotic mapping they were modified as shown in Figure 3.8:

- 'Unknown' cells are considered to be 'occupied' with the exception that their base value is '1' and not '0'. This modification is necessary because 'unknown' cells may be 'occupied', and the differences between the border cells (cells that are adjacent to 'unknown' cells) may be local inaccuracies. The base value is set to '1' not '0' as a compensation for the uncertainty of the actual cell value.

- When the distance grid is acquired, the 'unknown' cells are assigned value '-1'. It is done so that two cells wouldn't be evaluated as similar or dissimilar, when the actual cell value is unknown.



2	2	3	3	4	4	3	2	-	2	3	3	3	2	1	-
-	1	2	2	3	4	4	3	2	3	3	2	2	1	0	-
-	0	1	1	2	3	4	4	3	3	2	1	1	0	0	-
-	0	0	0	1	2	3	4	3	2	1	0	0	0	0	-
-	-	-	-	-	2	3	4	3	2	1	0	0	0	0	-
-	-	2	2	2	3	4	5	4	3	2	1	1	1	0	-
-	2	2	3	3	3	3	4	5	4	3	2	2	2	-	-
-	1	1	2	3	2	2	3	4	5	4	3	2	-	-	-

-	0	0	1	2	1	1	2	3	4	4	3	2	-	-	-
-	0	0	0	1	0	0	1	2	3	3	2	1	0	-	-
-	-	0	-	-	0	0	0	1	2	2	1	0	0	-	-
-	-	-	-	-	-	-	0	1	2	2	-	0	-	-	-

Figure 3.8. An example of a distance grid adapted for map merging

When the distance grids are computed, the algorithm uses two counters for the computation of 'occupied' cell similarity – 'sim' for similar cells and 'dis' for dissimilar cells (Figure 3.9). Similarly the 'free' cell similarity is computed.

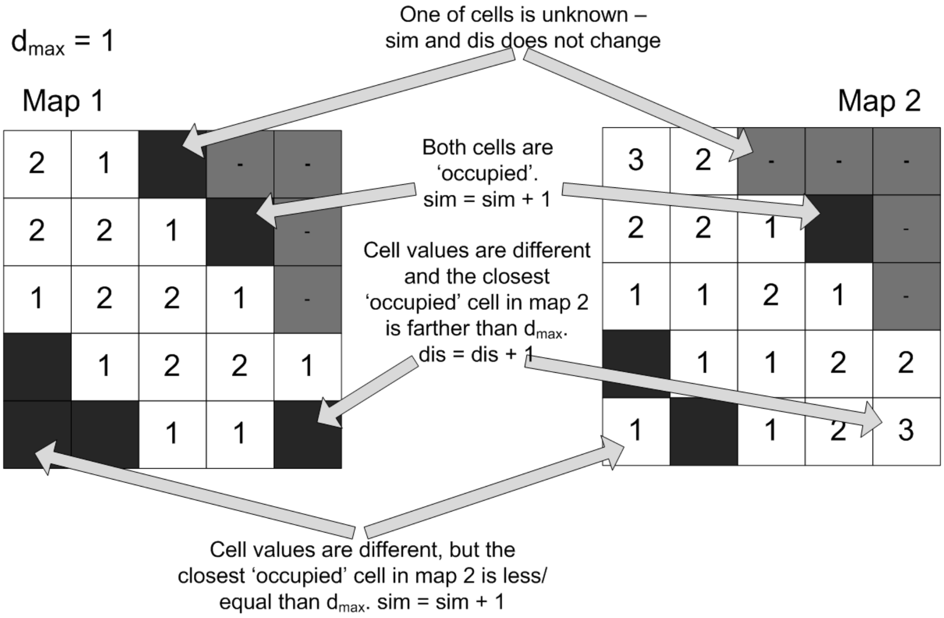


Figure 3.9. The computation of 'occupied' cell similarity

When all the cells are compared, the 'occupied' cell similarity is computed as a ration between similar cells and all the cells that didn't have a value 'unknown' in either map (equation 3.7):

$$s_{occ} = \frac{sim}{sim + dis} \tag{3.7.}$$

s_{occ} - 'occupied' cell similarity, *sim* – similar cell count, *dis* – dissimilar cell count

Equation 3.7. only computes the similarity of one map against the other. To compute the full 'occupied' cell similarity, the similarity of map M_1 against map M_2 and the similarity of map M_2 against map M_1 are computed, and then the average value of both numbers is computed (equation 3.8).

$$s_{occ\_tot} = \frac{s_{occ\_M1\_M2} + s_{occ\_M2\_M1}}{2} \tag{3.8.}$$

s_{occ_tot} – total 'occupied' cell similarity, $s_{occ\_M1\_M2}$ – 'occupied' cell similarity for map M_1 against map M_2, $s_{occ\_M2\_M1}$ – 'occupied' cell similarity for map M_2 against map M_1

3.4. The summary

In this chapter the method developed by the author of thesis for reliable robot map merging is described. This method can be used for dynamic proposal and rejection of map merging hypotheses and without losing information that is acquired after the map merging. the representation of the hypotheses as a hierarchical hypothesis tree reduced the computational and memory resources necessary for the map merging process. By using this representation, it is not necessary to simultaneously maintain local maps and global maps of all levels. Instead the global map can be acquired at any time by using the hypothesis tree and local maps.

The proposed method consists of two parts, each of which fulfils an important role in the creation and maintenance of the hypothesis tree. The global map merging and hypothesis maintenance part oversees the creation of the global map and checks, whether the hypothesis tree is correct. The local map merging searches for transformations between two maps by taking into account the previous experience of map mergings.

Although the developed map merging method is developed and evaluated (see chapter 5) for a particular map type – metric occupancy grid maps – it may be adapted and used for other map types with a condition that the relative positioning of the maps can be described in the form defined in chapter 3.1. To use the method for other map types the following components must be changed: the algorithm for local map merging and the evaluation algorithm of map merging hypothesis. Other components of the map merging method (the maintenance algorithms of hypothesis tree, rejected hypothesis list and map merging history) can remain unchanged even if map type changes.

4. THE IMPLEMENTATION OF MAPPING SYSTEM

In this chapter the mapping approach is developed and described. The map merging approach is based on the binary Bayes filter mapping [Thr 2005] (equation 4.1.). The modifications made the binary Bayes filter mapping allows using close range sensors.

$$l_t = l_{t-1} + \log\frac{p(x|z_t)}{(1 - p(x|z_t))} - \log\frac{p(x)}{1 - p(x)} \tag{4.1.}$$

l_t – posterior belief log odds for binary state variable, l_{t-1} – the previous log odds value of the cell, z_t – the sensor measurement value (‘occupied’ or ‘free’), $p(x|z_t)$ – probability that the cell is ‘occupied’ based on current measurement, $p(x)$ – the previous occupancy probability of the cell.

4.1. The description of the multi-robot system

For the implementation of the multi-robot system iRobot Roomba 560 vacuum cleaner robots are used [iRo 2012] that are additionally equipped with on Intel Atom CPU based computing device, WiFi and video camera [RTU 2013]. The goal of the multi-robot system is to provide effective coordination between robots to achieve high efficiency in task planning, task assignments and path planning. The WiFi and additional computing resources are necessary for the robot coordination and navigation, and the cameras are used for the localization purposes [RTU 2013].

Robots are capable to determine their location in the environment by using artificial landmarks. Only built-in collision and close range sensors, that can detect obstacles at the distance of few centimetres, are available for obstacle identification [iRo 2012].

The system is intended for use in indoor environment with previously known environment dimensions. The indoor environment has a high probability of change, and an automatic environment mapping is necessary for a robotic system to perform autonomously.

The problem considered in this chapter is the mapping with primitive sensors – collision sensors. In the developed robotic system the camera is directed to the ceiling and is only used for the robot localization. Only close range sensors are available for obstacle detection purposes. The only way to acquire the information about the environment with such sensors is to continuously compute the space that is taken by the robot and to register collisions with obstacles.

4.1.1. Localization with artificial landmarks

The multi-robot system uses an integrated indoor robot localization approach, that is based on the use of artificial landmarks – glyphs [Nik 2013]. The robots are able to visually track these landmarks.

To provide the robots with ability to determine their locations, the glyphs are attached to the ceiling, and robots use cameras to determine their locations against these glyphs. Each glyph is unique and recognizable from every side. By using this solution, robot coordinates and direction is always available, whenever the glyphs are visible.

Unfortunately because of the environment configuration it is not always possible to see glyphs. Odometry is another important information source, that is used for the localization, when the glyphs are not available.

4.1.2. Map representation

The maps in multi-robot system are represented by using occupancy grids. The position of every robot is represented as a triplet <*x, y, direction*>, where *x* is the X coordinate of robot, *y* is the Y coordinate of robot, and *direction* is the robot rotation.

4.2. Binary Bayes filter mapping

A simple but effective algorithm for the update of map cell occupancy values is the binary Bayes filter (equation 4.1.) [Thr 2005]. It is used in mapping to evaluate the binary state of the cells during mapping, where the binary state is the cell state (‘occupied’ = 1, and ‘free’ = 0). Independently of the current occupancy values, it is assumed that each cell can only have on correct state and it does not change during time, i.e. the environment is static. The binary Bayes filter maintains ‘a memory’ about the previous sensor measurements and at the same time ensures that the cell value is always in the interval [0; 1], that corresponds to the occupancy grid representation considered in the thesis.

4.3. Modified binary Bayes filter mapping

Unfortunately problems of practical nature do not allow using binary Bayes filter for this particular multi-robot system, therefore in this chapter modified binary Bayes filter mapping is offered.

Robots in the mapping system communicate with the server by using wireless network, and the maps are merged on the server. The resolution of the maps equals to 10×10 centimetres for cell to reduce the wireless network load. This cell size also means smaller map size and less computation necessary for mapping, map merging and path planning.

Due to the large cell size and poor robot sensor equipment modifications in the binary Bayes filter are necessary. In the normal case binary Bayes filter changes cell occupancy values equally for both 'occupied' and 'free' measurements. This situation is undesirable in this particular mapping system because of the way the robots acquire information about the environment – the map is updated by registering collisions with the obstacles and computing the space occupied by the robot without collisions. Usually the sensor measurements mark cells as 'occupied' only a couple of times – at the moment, when robot meets obstacle and close range sensors are activated. On the other hand, the cells are marked as 'free' at every position update, when a direct contact with the obstacle is not observed.

If the original Bayes filter is left unchanged, then the 'free' cell sensor measurements can quickly overwrite cells with value 'occupied'. As a result many cells that actually correspond to the occupied areas are incorrectly identified as 'free'.

To avoid this, the cell occupancy values are updated by using binary Bayes filter algorithm only in cases, when the sensor measurement is 'occupied'. If the cell is marked as 'free' in current sensor measurement, then its log odds value is decreased by a constant value, that is a function of sensor measurement frequency and the required cell value change time. The algorithm of the cell updates can be seen in Figure 4.1.

Cell value update (l_{t-1}, z_t):

$$l_t = \begin{cases} l_t = l_{t-1} + \log\frac{p(x|z_t)}{(1-p(x|z_t))} - \log\frac{p(x)}{1-p(x)}, & \text{if } z_t \text{ is occupied} \\ l_t = l_{t-1} - \text{LogOddsDecrease}, & \text{if } z_t \text{ is free} \end{cases}$$

$$p(x) = 1 - \frac{1}{1+\exp(l_t)}$$

$l_{t-1}=l_t$

return p(x)

Figure 4.1. Modified binary Bayes filter mapping algorithm

l_t represents the posterior belief log odds for the binary state variable. l_{t-1} is the previous log odds value of the cell. z_t is the value of the sensor measurement ('occupied' or 'free'). $p(x|z_t)$ is the probability that the cell value is 'occupied' based on the current measurement. $p(x)$ is the previous cell occupancy probability.

The constant *LogOddsDecrease* represents the speed of the decrease of the cell's log odds, and its value depends on the sensor measurement frequency. In the particular mapping system the time, in which the cell value should change from 'occupied' to 'free', was experimentally determined to be equal to 2.5 seconds, the sensor measurement frequency is 50 times per second, and the *LogOddsDecrease* value computed from these parameters equals 0.008 (equation 4.2.).

$$\mathrm{LogOddsDecrease} = \frac{1}{t * f} \tag{4.2.}$$

t – time, in which the cell value must change from ‘occupied’ to ‘free’ (seconds),
f – sensor measurement frequency (times/second)

4.4. The evaluation of map merging system

To test the ability of map merging system to create the environment map, an experiment in indoor environment was performed. The size of the environment is 3.5×3.5 metres that corresponds to 35×35 cells in an occupancy grid. The environment contains five obstacles – three boxes and two table legs.

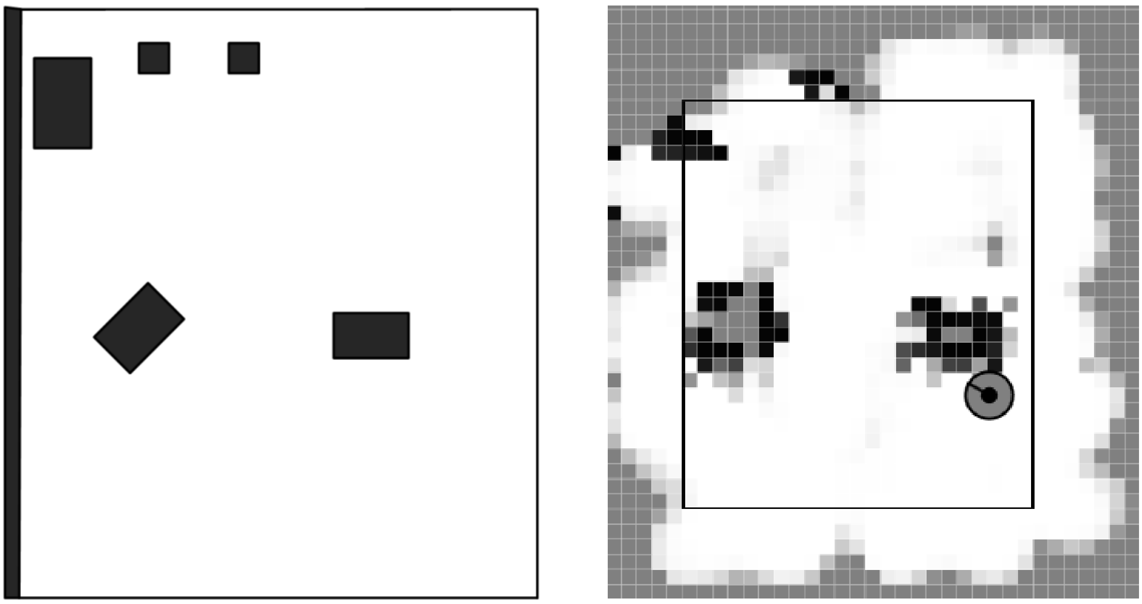


Figure 4.2. The configuration and the map of the environment

With the developed multi-robot system the environment was explored and the map seen in the Figure 4.2 was created. The black rectangle represents the virtual wall that is used to force the robot to remain in the test environment. If the robot tries to leave the marked are, it receives the signal to return. In figure 4.2 it can be seen that the robot has created a map that approximately corresponds to the actual situation and is usable for the navigation in the environment.

4.5. The summary

The problem considered in this chapter is the mapping with primitive sensors – collision sensors or very short range sensors. The only way to acquire the information about the environment with such sensors is to continuously compute the space that is occupied by the robot and to register the collisions or close range detections of obstacles.

Due to the glyph localization, the locations of the robots are always known and the values of the cells can be assigned according to the measurements of the sensors. A simple but effective algorithm for the updating of map cell occupancy values is the binary Bayes filter [Thr 2005]. The binary Bayes filter maintains 'a memory' about the previous sensor measurements and at the same time ensures that the value of the cell is always in interval [0; 1] that corresponds to the occupancy grid representation used in the thesis.

Unfortunately problems of practical nature do not allow using binary Bayes filter for this particular multi-robot system. If the original Bayes filter is left unchanged, then the 'free' cell sensor measurements can quickly overwrite cells with value 'occupied'. As a result many cells that actually correspond to the occupied areas are incorrectly identified as 'free'.

For this reason a modified binary Bayes filter mapping algorithm is offered in this chapter. Cell occupancy values are updated by using binary Bayes filter algorithm only in cases, when the sensor measurement is 'occupied'. If the cell is marked as 'free' in current sensor measurement, then its log odds value is decreased by a constant value, that is a function of sensor measurement frequency and the required cell value change time.

The results of experiments demonstrate that the proposed approach is capable of creating robot maps, which can be used by robots to navigate the environment and to complete tasks in specific places of environment.

5. THE IMPLEMENTATION OF MAP MERGING SYSTEM

This chapter describes the experimental software system, performed experiments and their results that allow to evaluate the performance of the developed map merging system, its effectiveness and practical applications.

5.1. The description of the experimental software system

The proposed map merging method ReMMerg is implemented in a software system, and the maps created by the multi-robot system that is described in chapter 4 are used as the input data. The developed software system completely implements all data structures and processes described in chapter 3. Additionally map merging log is created that enables the system's user to follow the events that take place in the system.

5.2. The selection of map merging hypothesis evaluation parameters

The goal of experiment:

The goal of the experiment is to determine the most appropriate parameters for the hypothesis evaluation method EvaLIM for the maps used in the experiments. The EvaLIM is compared with the most popular evaluation method in literature – direct cell comparison [Bir 2006], which is a special case of EvaLIM by the distance threshold $d_{max}=0$.

The implementation of experiment:

Both map merging hypothesis evaluation methods (EvaLIM and direct cell comparison) were compared with maps created in three different environment configurations. In environment configuration no.1 robots created eight partial maps; in the configurations no.2 and no.3 nine partial maps were created. The map merging process was tracked by sequentially saving partially created maps. Five mapping stages of each partial map were used for the experiments. An creation sequence of one partial map can be seen in Figure 5.1.

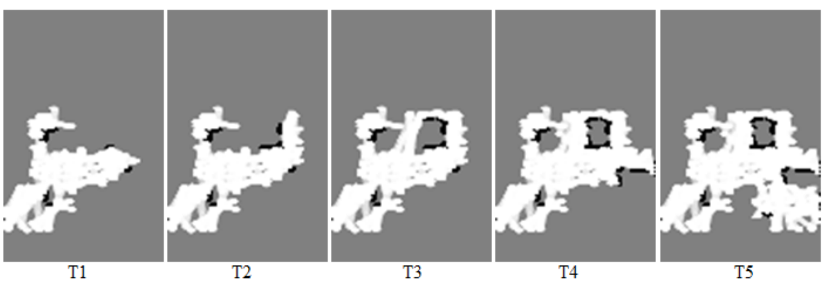


Figure 5.1. Example of partial map mapping sequence (T1, T2 etc. represent the time)

With every map set 36 global map merging attempts were made (104 global map creation attempts in total) – all possible combinations of three parameters:

- Three hypothesis confirmation thresholds $hval_{min}$ [0.93, 0.95, 0.97].
- Four hypothesis evaluation distance thresholds d_{max} [0, 1, 2, 3]. When the distance threshold $d_{max}=0$, the results returned by the EvaLIM method are identical to the direct cell comparison.
- Three local merging sets hset [8, 16, 24], which represent how many transformations are computed by local map merging method in a merging attempt.

For all map merging method parameter configurations the first 10 map merging steps are considered (more steps do not return better results – more successful mergings). Merging count in every step is equal to [n – 1], where n is the highest level hypothesis count.

The results of experiment:

The experiment results in Table 5.1. represent the average map count (percentage) in the largest hypothesis for each parameter configuration. The more maps are included in the largest hypothesis, the better the result is.

Table 5.1. Average map count (percentage) in the largest hypothesis for each configuration

		Distance threshold d_{max}											
		0			1			2			3		
		Hypothesis confirmation threshold $hval_{min}$											
		0,93	0,95	0,97	0,93	0,95	0,97	0,93	0,95	0,97	0,93	0,95	0,97
Local map set hset	8	0	0	0	45,37	18,51	7,4	49,07	52,77	45,37	45,83	45,37	37,96
	16	0	0	0	60,18	18,51	7,4	64,81	64,81	37,96	68,05	68,05	56,48
	24	0	0	0	49,07	25,92	7,4	68,51	68,51	60,18	60,64	79,16	71,29

No hypotheses are proposed for the distance threshold $d_{max}=0$ in any parameter configuration and map sets. The results returned by the EvaLIM with the distance threshold $d_{max}=0$ is identical to the direct cell comparison. The poor results for the threshold $d_{max}=0$ demonstrate that the direct cell comparison is not suitable for the evaluation of locally inaccurate maps and it is better to use EvaLIM evaluation method with distance threshold d_{max} that is higher than 0. The best results for the particular map set were acquired by distance thresholds $d_{max}=2$ and $d_{max}=3$ (respectively average 68,51% and 79,16% of maps are included in the largest hypothesis).

The method is able to create larger hypotheses by the distance threshold $d_{max}=3$, however, table 5.2. shows that the distance threshold $d_{max}=3$ allows incorrect mergings in many cases. In total, 62.96% of hypotheses by this distance threshold contain at least one incorrect hypothesis. These results show that larger global maps can be acquired with higher distance thresholds, but the risk of incorrect mergings is also considerably higher.

Table 5.2. The incorrect mergings (percentage from merging count) for all configurations

		Distance threshold d_{max}											
		0			**1**			**2**			**3**		
		Hypothesis confirmation threshold											
		0,93	**0,95**	**0,97**	**0,93**	**0,95**	**0,97**	**0,93**	**0,95**	**0,97**	**0,93**	**0,95**	**0,97**
Local merging set	**8**	0	0	0	0	0	0	0	0	0	66,66	66,66	33,33
	16	0	0	0	0	0	0	0	0	0	100	100	33,33
	24	0	0	0	0	0	0	0	0	0	66,66	66,66	33,33

To set the distance threshold in a robot system, two values should be considered: a) the noise in maps in terms of arbitrary cell position on map and in reality and b) cell size. The d_{max} value should be approximately equal to the distance of most errors in terms of cell size. From several possible values the authors of this paper recommend to use the highest distance threshold that yields acceptable level of incorrect mergings. In this particular robot system the distance threshold $d_{max}=2$ should be used.

The interpretation of results and conclusions

The experimental results show that the proposed global map merging method can create partial global maps and in some cases full global maps from multiple local maps, but the parameters must be chosen carefully for the best performance. The use of higher distance thresholds can achieve larger global maps, but the risk of wrong mergings is also higher.

Further in thesis the following map merging method parameters are used: Distance threshold $d_{max}=2$; Map merging hypothesis confirmation threshold $hval_{min}=0.95$; Local merging set $hset=16$.

5.3. Evaluation of ReMMerg performance

5.3.1. Global map creation

The goal of experiment:

The goal of the experiment is to evaluate the ability of the method to create the global map without any information about the relative coordinate systems and common areas of partial maps.

<u>The implementation of experiment:</u>

The input data are acquired with the experiments described in chapter 5.2. Only the results with previously chosen method parameters are considered. The results for each environment configuration are analyzed individually.

For each environment configuration the local merging possibility sets were created. Local merging possibility sets represent the combinations of local maps, which the local map merging method can merge by finding the transformation hypothesis between two maps, where the hypothesis evaluation value exceeds the hypothesis confirmation threshold.

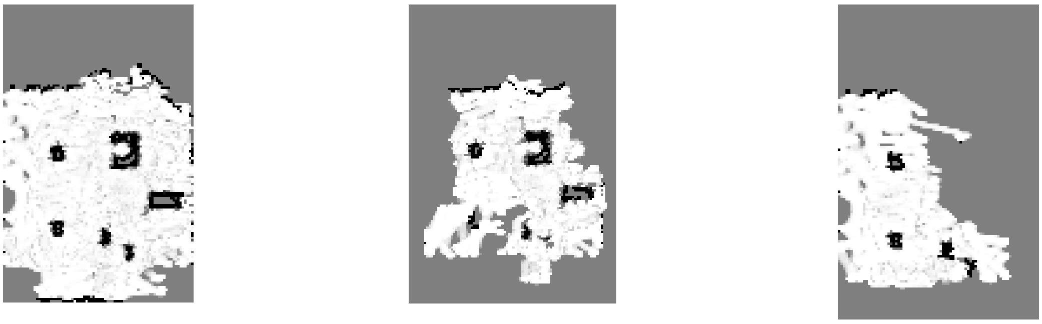
<u>The results of experiment (environment configuration no.1):</u>

In environment configuration no.1. the ReMMerg has proposed two highest level hypothesis (their comparison with full global map is depicted in Figure 5.2.) – **[[M3+M6]+[M1+M2]]** and **[M4+M7]**. Two maps (**M5** and **M8**) are not included in any hypotheses.

The largest highest level hypothesis contains only 50% of local maps, which is a poor result, but several factors that are independent of global map merging method do not allow to achieve better results:

- **Limitations of local map merging method**. None of the existing local map merging methods is able to find the correct common areas of maps in all cases.
- **Global inaccuracies in one of the maps**. Even though, as can be seen in local merging possibility sets in Figure 5.3., one set includes maps **M1, M2, M3, M5, M6**, only 4 maps are included in hypothesis **[[M3+M6]+[M1+M2]]**. It can be explained with the fact that a part of local map **M3** is globally inaccurate, and it does not allow to merge **[[M3+M6]+[M1+M2]]** (with map **M3**) with map **M5**, because their common parts are too different.

Full global map	ReMMerg hypothesis no.1	ReMMerg hypothesis no.2



[[M3+M6]+[M1+M2]] [M4+M7]

Figure 5.2. Environment configuration no. 1: manually created full global map (left side) and the highest level hypotheses created by ReMMerg (right side)

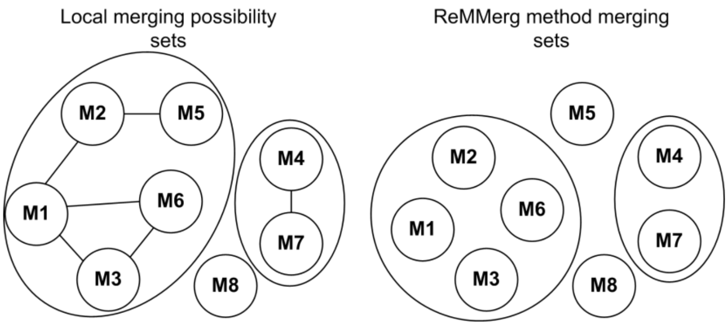


Figure 5.3. Environment configuration no. 1: local merging possibility sets

Taking into account the factors mentioned before, it can be concluded that ReMMerg method has achieved the best results that are possible by the accuracy of the local maps and limitations of the local map merging method.

The results of experiment (environment configuration no.2):

In environment configuration no.2. the ReMMerg has proposed one global map hypothesis (its comparison with full global map is depicted in Figure 5.4.) – **[M9+[[M2+[M4+M8]]+[M7+[M6+[M1+M5]]]]**. **M3** is not included in the hypothesis.

Full global map **ReMMerg hypothesis**

Figure 5.4. att. Environment configuration no. 2: manually created full global map (left side) and the global map hypothesis created by ReMMerg (right side)

The only map that is not included in the global map hypothesis is **M3**. Map M3 accordingly to local merging possibility set (Figure 5.5.) can be merged with two other maps – **M1** and **M6**. No global inaccuracies are present in any of local maps of environment configuration no. 2.

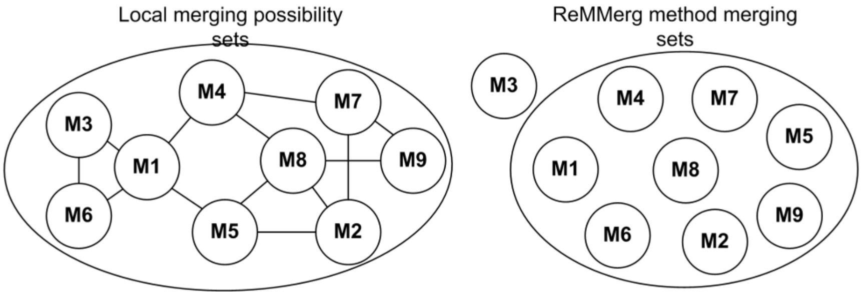


Figure 5.5. Environment configuration no. 2: local merging possibility sets

<u>The results of experiment (environment configuration no.3):</u>

In environment configuration no.3. the ReMMerg has proposed two highest level hypotheses (their comparison with full global map is depicted in Figure 5.6.) – **[M4+M6]** and [[**M5+M8]+[M9+[M7+[M2+M3]]]]**. Map **M1** is not included in any hypotheses.

Full global map **ReMMerg hypothesis no. 1** **ReMMerg hypothesis no. 2**

[M4+M6] **[[M5+M8]+ [M9+[M7+[M2+M3]]]]**

Figure 5.6. Environment configuration no. 3: manually created full global map (left side) and the highest level hypotheses created by ReMMerg (right side)

The comparison of local merging possibility sets and the results acquired by ReMMerg (Figure 5.7.) demonstrate that the two highest level hypotheses **[M4+M6]** and [[**M5+M8]+[M9+[M7+[M2+M3]]]]** are subsets of the local merging possibility set. Map **M1** is not included in any hypothesis, and it cannot be directly merged with any other local map.

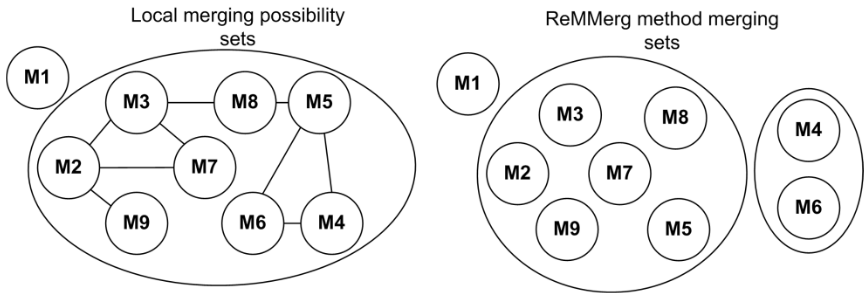


Figure 5.7. Environment configuration no. 3: local merging possibility sets

The interpretation of results and conclusions:

The full global map was not created in any of the considered map merging cases. In all environment configurations, as can be seen in local merging possibility sets, larger hypotheses are possible. Only in the environment configuration no. 1 one hypothesis is unachievable because of a globally inaccurate map.

5.3.2. Proposal of global map hypothesis

The goal of experiment:

The goal of the experiment is to verify, if the ReMMerg method proposes the largest possible hypotheses by the existing inaccuracies of local maps and local map merging method limitations.

The implementation of experiment:

To verify, if the ReMMerg method proposes the largest possible hypotheses, all possible map merging hypotheses for all environment configurations were computed.

The results of experiment (environment configuration no.1):

Table 5.3. depicts, how often specific local maps are encountered in different hypothesis levels in environment configuration no.1, where the levels represent the count of local maps included in the hypotheses.

Table 5.3. Environment configuration no.1: Map count in hypotheses (%)

	M1	M2	M3	M4	M5	M6	M7	M8
2	40	40	40	20	20	20	20	0
3	0	100	100	0	0	100	0	0
4	100	100	100	0	0	100	0	0
5	0	0	0	0	0	0	0	0
6	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0	0
Total	**42,86**	**57,14**	**57,14**	**14,28**	**14,28**	**42,86**	**14,28**	**0**

According to the data, the largest hypothesis, which can be created in environment configuration no.1 with Hough transformation map merging [Car 2008], contains four local maps – **M1**, **M2**, **M3** and **M6**. Map **M8** is not included in any hypotheses but maps **M4**, **M5** and **M7** appear only in hypotheses that contain two local maps.

It can be concluded that the ReMMerg method achieves the best possible results in the environment configuration no. 1 (two hypotheses **[[M3+M6]+[M1+M2]]** and **[M4+M7]**), which are possible by the local map merging method and chosen method parameters.

<u>The results of experiment (environment configuration no.2):</u>

The Table 5.4. and Figure 5.8. depicts, how often specific local maps are encountered in different hypothesis levels in environment configuration no.2, where the levels represent the count of local maps included in the hypotheses.

Table 5.4. Environment configuration no.2: Map count in hypotheses (%)

	M1	M2	M3	M4	M5	M6	M7	M8	M9
2	36,36	27,27	18,18	18,18	27,27	18,18	18,18	27,27	9,09
3	55,17	41,37	13,79	41,38	44,83	20,69	17,24	48,27	17,24
4	74,44	52,22	14,44	56,67	65,55	34,44	18,89	58,89	24,44
5	85,66	61,09	20,82	71,67	75,43	48,8	24,23	73,04	39,25
6	93,83	71,65	30,97	81,23	83,73	63,52	31,76	87,4	55,9
7	97,2	84,59	44,65	91,41	89,5	77,44	46,08	94,48	74,64
8	98,98	93,22	68,16	99,52	94,24	91,78	67,15	96,94	89,99
9	100	100	100	100	100	100	100	100	100
Total	**96,40**	**86,02**	**58,04**	**91,94**	**90,49**	**81,28**	**58,52**	**93,04**	**78,09**

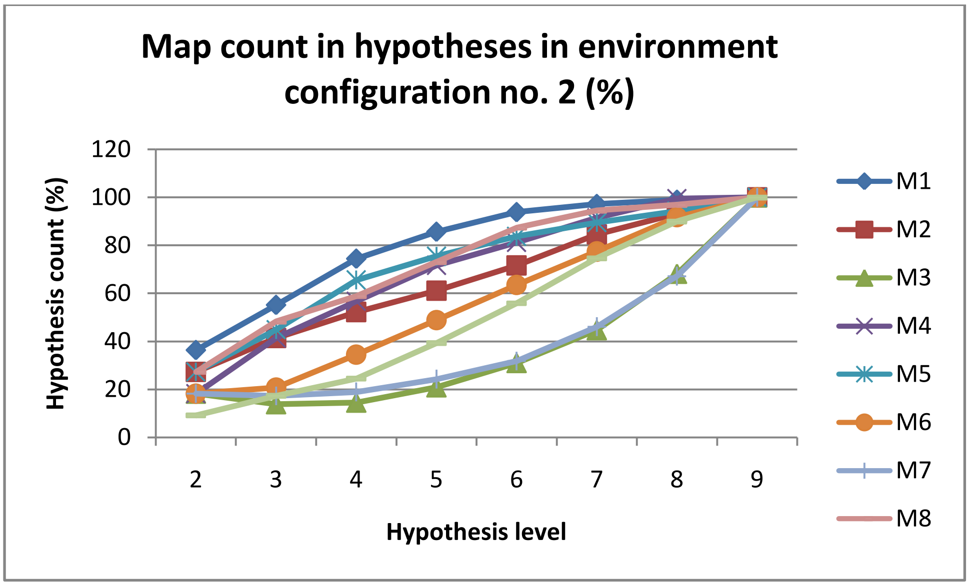


Figure 5.8. Environment configuration no.2: Map count in hypotheses (%)

Even though there are paths to propose hypotheses that contain 9 local maps, ReMMerg has proposed a hypothesis that only contains 8 maps – **[M9+[[M2+[M4+M8]]+[M7+[M6+[M1+M5]]]]** and which does not include map **M3**. It can be seen in Table 5.4. that the local map **M3** is included in the hypotheses the least often (only 58.04% hypotheses).

<u>The results of experiment (environment configuration no.3):</u>

The Table 5.5. and Figure 5.9. depicts, how often specific local maps are encountered in different levels in environment configuration no.2. The data shows that it is possible to propose hypotheses from 9 local maps in environment configuration no.3. Instead of the best result, the ReMMerg has proposed hypotheses from 2 and 6 local maps – **[M4+M6]** and **[[M5+M8]+[M9+[M7+[M2+M3]]]]**. None of these hypotheses contain map M1, which is the rarest map in hypotheses (it is encountered only in 32.01% hypotheses).

Table 5.5. Environment configuration no.3: Map count in hypotheses (%)

	M1	M2	M3	M4	M5	M6	M7	M8	M9
2	0	33,33	33,33	22,22	33,33	22,22	22,22	22,22	11,11
3	6,67	33,33	33,33	33,33	53,33	20	40	40	40
4	9,09	33,33	42,42	45,45	75,76	24,24	51,51	75,75	42,42
5	8,11	60,81	48,65	55,4	91,89	27,03	70,27	87,84	50
6	16,49	73,94	64,36	71,28	100	40,96	76,59	93,62	62,76
7	24,37	86,55	79,55	84,59	100	61,06	85,99	97,48	80,39
8	41,77	98,17	89,63	92,99	100	86,89	95,43	100	95,12
9	100	100	100	100	100	100	100	100	100
Total	**32,01**	**84,36**	**77,28**	**81,60**	**97,52**	**64,03**	**85,00**	**95,03**	**78,93**

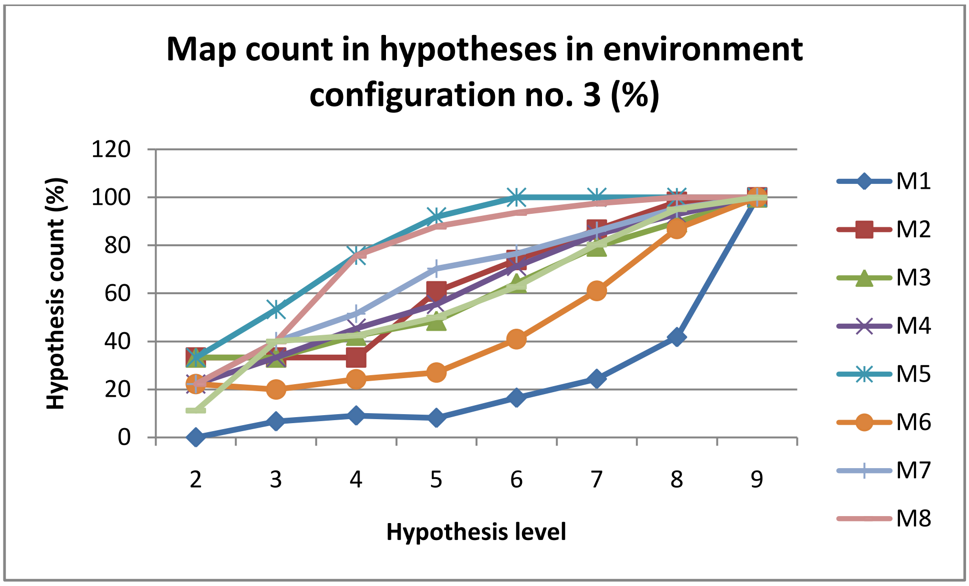


Figure 5.9. Environment configuration no.3: Map count in hypotheses (%)

<u>The interpretation of results and conclusions:</u>

After the comparison of the ReMMerg results with the full hypotheses set in each configuration, it can be concluded that not always the best possible results have been achieved. In environment configuration no.2 and no.3 the global map merging method ReMMerg reaches deadlock because of the chosen map merging paths – all proposed hypotheses are acceptable but can't be merged. This situation has two explanations:

- Local map merging method can merge local maps but is not able to merge all higher level hypotheses.
- Local map merging method can find the correct transformation between the highest level hypotheses but due to higher noise level the proposed hypothesis has low evaluation value and it is not accepted.

5.3.3. Proposing hypotheses with adaptive hypothesis confirmation threshold

The goal of the experiment:

The goal of the experiment is to test, whether the ReMMerg method can propose the largest possible hypotheses or at least to improve the previous results by making changes in the application of hypothesis confirmation threshold.

The implementation of experiment:

For the implementation of the experiment a modified software system was used, where the changes in hypothesis confirmation threshold usage $hval_{min}$ were implemented. For the confirmation of two local map merging the set threshold $hval_{min}=0.95$ is used, but, when more than two local maps are involved in merging, the hypothesis confirmation threshold is reduced accordingly to the sum of local maps in both merging components (Table 5.6).

Table 5.6. The changes in hypothesis confirmation threshold accordingly to local map count

Local map count	Hypothesis confirmation threshold $hval_{min}$	Example
2	0.95	M1 un M2
3	0.948	M1+2 un M3
4	0.946	M1+2 un M3+4 || M1 un M2+M3+M4
5	0.944	M1+2 un M3+M4+M5
6	0.942	M1+M2+M3 un M4+M5+M6
7	0.940	M1+M2+M3 un M4+M5+M6+M7
8	0.938	M1+M2+M3 un M4+M5+M6+M7+M8
9	0.936	M1+M2+M3+M4+M5 un M6+M7+M8+M9

With these changes in the usage of hypothesis confirmation threshold $hval_{min}$ the repeated global map merging was performed for all three environment configurations by the method parameters $[d_{max}=2;\ hval_{min}=0.95;\ hset=16]$.

<u>The results of experiment:</u>

When the repeated global map merging was performed in environment configuration no.1, the changes in hypothesis confirmation threshold do not change the result – the proposed hypotheses. Repeated global map merging in environment configuration no.2 return different hypothesis tree (hypotheses and their maps can be seen in Figure 5.10.), however, only 8 out of 9 maps are included in the hypothesis (in both cases the map **M3** is not included).

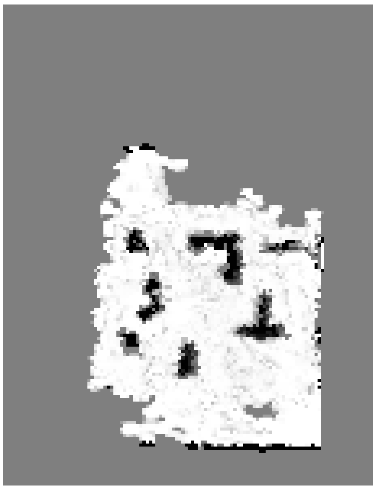
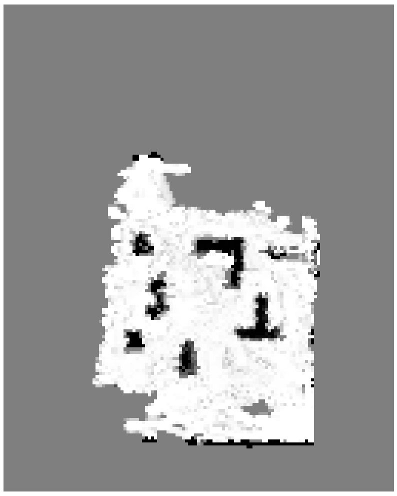
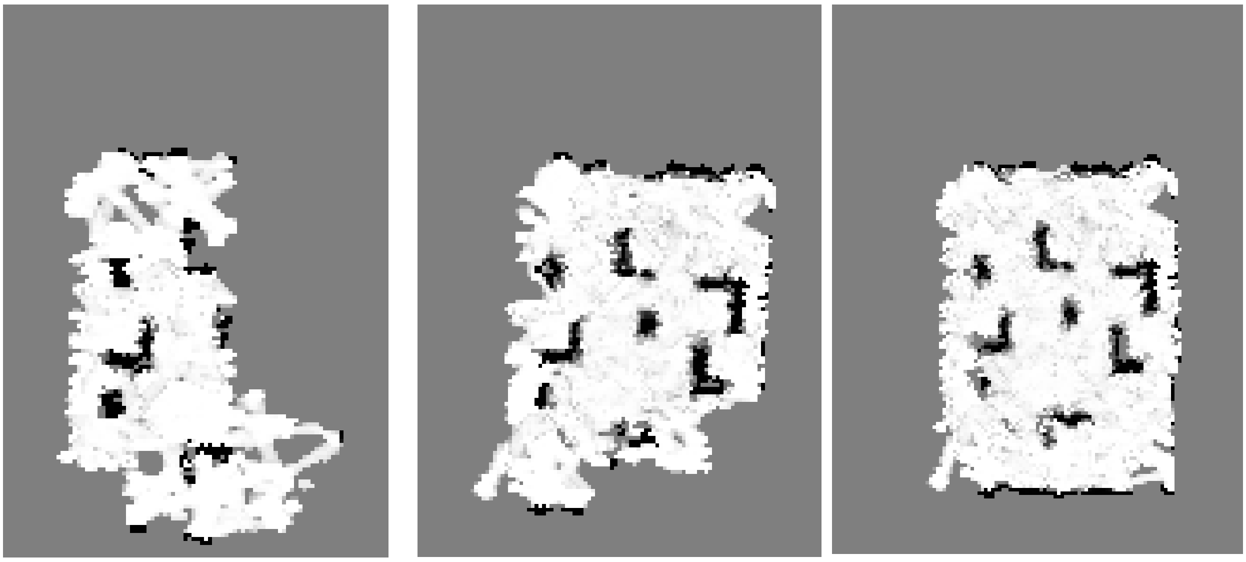
Initial hypothesis	**Hypothesis after changes**
	
[M9+[[M2+[M4+M8]]+ [M7+[M6+[M1+M5]]]]	**[M9+[[M6+[M1+M5]]+ [M7+[M2+[M4+M8]]]]]**

Figure 5.10. Environment configuration no.2: Initial hypothesis (left side) and the hypothesis after the changes in hypothesis confirmation threshold (right side)

Unlike the first two environment configurations, in the environment configuration no.3 the changes in hypothesis confirmation threshold increases the local map count in the largest hypothesis to the largest possible count – 9 maps (Figure 5.11), which is a significant improvement when compared with initial results – hypotheses that contain 2 and 6 local maps.

These results confirm that the introduction of an adaptive hypothesis confirmation threshold improves the results of global map merging in one case – environment configuration no.3. To test, whether it is not just a coincidence, the data of all possible hypotheses in all environment configurations were analysed.

Initial hypothesis no. 1 **Initial hypothesis no. 2** **Hypothesis after changes**



[M4+M6] [[M5+M8]+[M9+[M7+[M2+M3]]]] [M1+[[M4+M6]+[[M5+M8]+[M9+[M7+[M2+M3]]]]]]

Figure 5.11. Environment configuration no.3: Initial hypothesis (left side) and the hypothesis after the changes in hypothesis confirmation threshold (right side)

Table 5.7. represents the count of hypotheses and the deadlock count/percentage in different hypothesis levels for environment configuration no.1, where the levels represent the count of local maps included in the hypotheses. Deadlocks represent the hypothesis count/percentage that are not included in any global map hypothesis – the hypotheses that include all local maps. As can be seen in Table 5.7. there are no changes in hypotheses and deadlock count after introduction of adaptive hypothesis confirmation threshold.

Table 5.7. Environment configuration no.1: Deadlocks before and after changes

Map count in hypothesis	Initial hypotheses	Initial deadlocks	Initial deadlocks (%)	Hypotheses after change	Deadlocks after change	Deadlocks after change (%)
2	5	5	100	5	5	100
3	1	1	100	1	1	100
4	1	1	100	1	1	100
5	0	0	0	0	0	0
6	0	0	0	0	0	0
7	0	0	0	0	0	0
Total	**7**	**7**	**100**	**7**	**7**	**100**

A different situation can be seen in Table 5.15., which represents the hypothesis count and deadlock count/percentage in different hypotheses levels for environment configuration no.2. The count of hypotheses has increased from 4320 to 5046 (hypotheses that include 9 local maps are not included in this count). The total deadlock count has also increased but its ratio against the total hypothesis count has decreased by 4.51%, which means that the probability to create a global map hypothesis from 9 local maps has increased.

Table 5.8. Environment configuration no.2: Deadlocks before and after changes

Map count in hypothesis	Initial hypotheses	Initial deadlocks	Initial deadlocks (%)	Hypotheses after change	Deadlocks after change	Deadlocks after change (%)
2	11	0	0	11	0	0
3	29	5	17,24	29	2	6,90
4	90	38	42,22	93	27	29,03
5	293	199	67,92	311	182	58,52
6	762	590	77,43	836	593	70,93
7	1467	1125	76,69	1719	1231	71,61
8	1668	1141	68,40	2047	1356	66,24
Total	**4320**	**3098**	**71,71**	**5046**	**3391**	**67,20**

The improvement from using adaptive hypothesis confirmation threshold is even more visible for the environment configuration no.3. in the Table 5.9., which represents the hypothesis count and deadlock count/percentage in different hypotheses levels for environment configuration no.3. it can be seen that the hypothesis count has increased from 1004 to 1053 and the total deadlock count has also increased. However, their ratio against the total hypothesis count has decreased by 33.96%, which means that the chance to reach a global map hypothesis that contains all 9 local maps, has significantly increased.

Table 5.9. Environment configuration no.3: Deadlocks before and after changes

Map count in hypothesis	Initial hypotheses	Initial deadlocks	Initial deadlocks (%)	Hypotheses after change	Deadlocks after change	Deadlocks after change (%)
2	9	2	22,22	9	0	0
3	15	5	33,33	15	0	0
4	33	19	57,57	33	3	9,09
5	74	59	79,73	74	27	36,49
6	188	157	83,51	192	98	51,04
7	357	310	86,83	375	232	61,87
8	328	258	78,66	355	132	37,18
Total	**1004**	**810**	**80,68**	**1053**	**492**	**46,72**

The interpretation of results and conclusions:

Many of the acceptable hypotheses are rejected due to not high enough hypothesis evaluation value, but in many cases this evaluation value is just marginally lower than the set hypothesis confirmation threshold $\mathtt{hval}_{\mathtt{min}}\mathtt{=0.95}$. The results demonstrate that the usage of adaptive hypothesis confirmation threshold improves the results (the local map count in the largest hypothesis) and decreases a chance to reach a deadlock (in environment configuration no.3 the deadlock hypothesis count decreases by 33.96%), but there are still a chance that the largest possible global map hypothesis will not be created. In real life situations, when the robots explore the environment simultaneously, the author suggests to address this problem in one of the following ways: to continue the exploration or to perform more map merging attempts by attempting to merge not only hypotheses but also their components.

5.3.4. Reversibility of the map merging

The goal of experiment:

The goal of the experiment is to test the ability of the method to continue the global map merging after the proposal of an incorrect hypothesis.

The implementation of experiment:

Experiment uses the data the acquisition of which are described in section 5.2. by performing 108 global map merging attempts for 36 different map merging method parameter combinations in three environment configurations.

The results of experiment:

There is always a possibility to propose an incorrect map merging hypothesis, and at some point during mapping it can be discovered that a previously proposed hypothesis is no longer acceptable. In this situation the map merging method must propose new hypotheses by taking into account previous experience.

The maps used in experiments represent only the last 20-40% of mapping process, but, as can be seen in Table 5.10, the amount of rejected hypotheses is considerably large (in one configuration as much as 50.43% of hypotheses are rejected, but 0% rejected hypotheses are only encountered in those configurations, where no hypotheses are proposed).

Table 5.10. Average count of rejected hypothesis (%)

		Distance threshold d_{max}											
		0			1			2			3		
		0,93	0,95	0,97	0,93	0,95	0,97	0,93	0,95	0,97	0,93	0,95	0,97
Local merging set	8	0	0	0	7,41	0	0	15	13,89	35,55	40,47	27,38	35,08
	16	0	0	0	7,41	11,11	16,67	26,67	11,43	36,9	38,65	36,41	23,61
	24	0	0	0	9,52	11,11	16,67	32,94	11,43	25,83	50,43	38,89	19,44

One way to address the problem of incorrect hypothesis proposals is to merge maps only when the exploration of the environment is finished. However, this approach does not let the robots to use the information acquired by the other robots during the exploration. Table 5.11., which represents the hypothesis percentage that were proposed before the end of mapping and not rejected, proves that it is possible to propose hypotheses in earlier mapping stages whose evaluations exceed the hypothesis confirmation threshold at the end of mapping.

Table 5.11. Average count of hypotheses proposed before the end of mapping and not rejected (%)

		Distance threshold d_{max}											
		0			1			2			3		
		0,93	0,95	0,97	0,93	0,95	0,97	0,93	0,95	0,97	0,93	0,95	0,97
Local merging set	8	0	0	0	26,19	16,67	0	68,25	54,92	50	51,67	58,33	80
	16	0	0	0	26,19	33,33	0	39,05	48,81	44,44	68,33	77,38	80,16
	24	0	0	0	68,89	27,78	0	50,79	50,79	68,25	30	79,17	82,74

The interpretation of results and conclusions:

All experiments in the chapter 5 are implemented by using robot maps that change in time. Results described previously show that by choosing appropriate ReMMerg method parameters it is possible to create at least partial global map hypotheses, which do not include incorrect mergings even if there were such mergings in earlier map merging stages.

The results of this particular experiment demonstrate that in most method parameter configurations at least some hypotheses were rejected. It means that there is a risk of incorrect mergings, and in these cases it is necessary to return into previous merging state.

Even though it is more complicated to merge maps before the end of mapping because of the incorrect merging risk, the results of experiments show that the benefits are significant – in many cases it is possible to propose correct map merging hypotheses before the end of mapping.

5.4. Summary

In this chapter the developed experimental system, the performed experiments and their results are described. The experimental results allow to evaluate the performance of map merging method, its effectiveness and practical applications.

6. CONCLUSIONS AND RESULTS

The **goal of the thesis** was to develop and implement a map merging method for the acquisition of the multi-robot system global map, that implements reversible and dynamic map merging during mapping.

To address the problem of dynamic and reversible map merging, the author has developed a map merging method ReMMerg. This method can be used for dynamic proposal and rejection of map merging hypotheses without losing information that is acquired after map merging. ReMMerg ensures that only acceptable hypotheses (their evaluation exceeds a previously set threshold) are proposed.

6.1. The evaluation of ReMMerg method

The map merging method was evaluated with robot maps created by a multi-robot system developed in Riga Technical university, and it was concluded that the method is capable of creating at least partial global maps, when the relative coordinate systems of the robots are unknown and locally inaccurate occupancy grids are used as maps. The method can recover from incorrect map mergings and create a new global map taking into account the previous map merging experience.

Hypothesis evaluation method EvaLIM

The results of experiments show that the proposed global map merging method can create partial global maps and in some cases full global maps, but the method parameters must be carefully chosen to achieve the best results. By comparing the developed hypothesis evaluation method EvaLIM with an approach most often used in literature – direct cell comparison [Bir 2006], which is a special case of EvaLIM by distance threshold $d_{max}=0$, it was concluded that EvaLIM perform much better in the case of noisy maps, if an appropriate distance threshold is set. To set the distance threshold d_{max} for a particular multi-robot system two values must be taken into account: a) noise in the maps – how much the obstacle position in map can differ from the real life situation and b) the size of map cells. The d_{max} value should be approximately equal to the distance of most errors in terms of cell size. From several possible values the authors of this paper recommend to use the highest distance threshold that yields acceptable level of incorrect mergings.

Dynamical creation of global map with ReMMerg

Initially, when the ability of the ReMMerg method to create a global map by the chosen parameters [d_{max}=2; $hval_{min}$=0.95; hset=16] was evaluated, a full global map was not created in any of the three environments. After the comparison of the ReMMerg results with the full hypotheses set in each configuration, it was concluded that not always the best possible results have been achieved (in two cases it was possible to create a global map hypothesis, and in one case the best possible result was achieved).

The main obstacle, which prevents the proposition of the largest possible hypothesis, is that the global map merging method reaches a deadlock – situation, when all the proposed hypotheses are acceptable but cannot be further merged. This situation has two explanations:

- Local map merging method can merge local maps but is not able to merge all higher level hypotheses.
- Local map merging method can find the correct transformation between the highest level hypotheses but due to higher noise level the proposed hypothesis has low evaluation value and it is not accepted.

With a goal to improve the performance of the ReMMerg method, changes in the application of hypothesis confirmation threshold were performed. The results demonstrate that the usage of adaptive hypothesis confirmation threshold improves the results (the local map count in the largest hypothesis) and decreases a chance to reach a deadlock, but there are still a chance that the largest possible global map hypothesis will not be created. In real life situations, when the robots explore the environment simultaneously, the author suggests to address this problem in one of the following ways: to continue the exploration or to perform more map merging attempts by attempting to merge not only hypotheses but also their components.

Reversibility of the map merging

There is always a possibility to propose an incorrect map merging hypothesis, and at some point during mapping it can be discovered that a previously proposed hypothesis is no longer acceptable. In this situation the map merging method must propose new hypotheses by taking into account previous experience.

The results of this particular experiment demonstrate that in most method parameter configurations at least some hypotheses were rejected. It means that there is a risk of incorrect mergings, and in these cases it is necessary to return into previous merging state.

Even though it is more complicated to merge maps before the end of mapping because of the incorrect merging risk, the results of experiments show that the benefits are significant –

in many cases it is possible to propose correct map merging hypotheses before the end of mapping.

Summary

The proposed global map merging method ReMMerg can create partial global maps and in some cases full global maps, but for the best results the method parameters must be carefully chosen. The longer the mapping is performed, the better is the chance that the full global map will be created.

The performance of the ReMMerg method is closely tied to the chosen local map merging method. If the local map merging method is not suitable for the local map merging, the ReMMerg method will not be able to create a global map. In the thesis a local map merging with Hough transformation was implemented [Car 2008]. This local map merging method is intended for the maps that contain a lot of straight lines. Consequently this approach and the global map merging method performs worse, if the maps do not contain enough straight lines. Instead of the local map merging method used in thesis any other occupancy grid merging method can be used without changing the global map merging method ReMMerg itself.

ReMMerg method can identify incorrect map mergings and return into previous state by rejecting all incorrect hypotheses, but for this task appropriate method parameters must be set. With appropriate method parameters it is possible to create at least partial global map hypotheses, which do not include incorrect mergings even if there were such mergings in earlier map merging stages.

Currently the ReMMerg method is intended for use in multi-robot systems with one central computing unit, which performs the global map merging. However, if the method were supplemented with a local hypothesis tree merging component, which solves the conflicts between different hypothesis trees, it would be possible to use the ReMMerg method in decentralized multi-robot systems.

6.2. Theoretical results

The following theoretical results were achieved during the thesis development:

- The map merging methods in multi-robot systems were surveyed and analysed and, based on analysis, map merging method characteristics were defined, that must be taken into account in map merging: the relative coordinate systems of the robots, map type, information used in map merging, map merging time and the precision of maps.

- Based on the map merging method analysis, two map merging aspects were identified and defined – local map merging and global map merging -, and their importance in map merging systems described. The local map merging addresses the search of transformation between two maps, their merging and the evaluation of the result. The global map merging takes into account, how often the maps are merged and how to choose maps for merging. In this case the opportunity arises to reject a map merging hypothesis, and to merge maps several times taking into account the previous experience.
- The necessity of the global map merging, when the robot positions are unknown, was demonstrated and experimentally proved.
- The reliable map merging concept was defined on the basis of local and global map merging concepts. Reliable map merging is a map merging, that ensures the reversibility of the map merging – it is possible to return to the state before the merging without losing information acquired after then merging in the local maps of the robots. The map merging decision is only made, if the hypothesis is believable - its evaluation exceeds an empirically set threshold.
- An occupancy grid map merging hypothesis evaluation algorithm EvaLIM was developed, that takes into account the local inaccuracies of the robot maps.
- <u>The main theoretical result</u> is the map merging method ReMMerg, that is developed on the basis of the reliable map merging concept. The method is noteworthy with the data structures and processes that provide the map merging reversibility – hypothesis representation, hypothesis trees, rejected hypothesis list and map merging history. As a result the method is a reversible and dynamic map merging method.

6.3. Practical results

The following practical results were achieved during the thesis development:

- The map merging software system was developed, that is based on the ReMMerg method and implements all its data structures and processes. It is experimentally proved that the method can create a global map, if all maps have common area with at least one other map, and the local map merging method is able to find this common area.

- A simple mapping algorithm, based on the binary Bayes filter mapping, was developed, that allows to create the environment map with short range sensors, if the positions are available.
- An occupancy grid map merging hypothesis evaluation method was developed, that takes into account the local inaccuracies of the maps. The comparison with the most common evaluation in the examined literature shows that the proposed evaluation performs significantly better, if the maps are locally inaccurate or the hypothesis have small but acceptable errors.

6.4. Further research

The possible further research are:

- The creation of individual hypothesis trees for each robot. Currently the developed method is intended for map merging, when the merging is performed by one central agent (robot or software agent). A hypothesis tree merging algorithm is necessary to use the method in a decentralized manner.
- The use of several local map merging methods in map merging. The local map merging part of ReMMerg is separated from the global map merging part. The local map merging may be performed with different methods in one mapping system. If the method would be able to detect, which local map merging method is the most appropriate for the particular map pair, it could significantly improve the effectiveness of the map merging method.

Bibliography

[Adl 2008] Adluru N. et. al. Merging maps of multiple robots. 19th International Conference on Pattern Recognition, IEEE, 2008, pp. 1-4.

[Aln 2010] G. Alnounou; Hudas, J. Overholt, and M. Krishnan, “Occupancy Grid Map Merging Using Feature Maps,” 2010.

[Ami 2005] Amigoni F., Gasparini S., Gini M. Merging Partial Maps Without Using Odometry. Multi-Robot Systems. From Swarms to Intelligent Automata. – Netherlands: Springer, 2005, III vol., pp. 133-144.

[And 2009] Andersone I. Exploration with multiple agents. Master’s thesis. – Riga, 2009, 83 p.

[Bal 2010] Ballesta M., Gil A., Reinoso O., Julia M., Jimenez L.M., „Multi-robot Map Alignment in Visual SLAM”, WSEAS Transactions on Systems, Volume 9, Issue 2, 2010, pp. 213-222.

[Bir 1996] Birk A. „Learning Geometric Concepts with and Evolutionary Algorithm”, Evolutionary programming V, The MIT Press, 1996

[Bir 2006] Birk A., Carpin S. Merging Occupancy Grid Maps From Multiple Robots. Proceedings of the IEEE. – IEEE, 2006, 94. vol., pp. 1384-1397.

[Bur 2002] Burgard W., Moors M., Schneider F. Collaborative Exploration of Unknown Environments with Teams of Mobile Robots. Lecture Notes in Computer Science. – Berlin/Heidelberg: Springer, 2002, 2466. vol., pp. 187-215.

[Bur 2005] Burgard W., Moors M., Stachniss C., Schneider F. Coordinated Multi-Robot Exploration // In IEEE Transactions on Robotics, vol.21, 2005, pp. 376-386.

[Car 2005] Carpin S., Birk A. Stochastic Map Merging in Rescue Environments. Lecture Notes in Computer Science. – Berlin/Heidelberg: Springer, 2005, 3276. vol., pp. 483-490.

[Car 2008] Carpin S. Fast and accurate map merging for multi-robot systems. Autonomous Robots. – USA: Kluwer Academic Publishers, 2008, 25. vol., pp. 305-316.

[Cec 2006] Ceccarelli N. et. al. Set Membership Localization and Map Building for Mobile Robots. Systems & Control: Foundations & Applications. – Boston: Birkhauser, 2006, Part III, pp. 289-308.

[Che 2007] Chen Z., Samarabandu J., Rodrigo R. Recent advances in simultaneous localization and map-building using computer vision, Advanced Robotics, Vol. 21, No. 3-4, 2007, pp. 233-265

[Ded 2000] Dedeoglu G., Sukhatme G. Landmark-based Matching Algorithm for Cooperative Mapping by Autonomous Robots. Distributed Autonomous Robotics Systems, October 2000. – USA: Springer, 2000, pp. 251-260.

[Fox 2000] D. Fox, W. Burgard, and H. Kruppa, “A probabilistic approach to collaborative multi-robot localization,” Autonomous Robots, vol. 8, no. 3, 2000.

[Guo 2008] R. Guo, Y. Li, and W. Chen, “Adaptive genetic algorithm for occupancy grid maps merging,” 2008 7th World Congress on Intelligent Control and Automation, pp. 5716–5720, 2008.

[Ho 2005] Ho K., Newman P. Multiple Map Intersection Detection using Visual Appearance. 3rd International Conference on Computational Intelligence, December 2005. – Singapore, 2005

[How 2006] Howard A. Multi-Robot Simultaneous Localization and Mapping using Particle Filters. The International Journal on Robotics Research. – Sage science press, 2006, 25. vol., No. 12, pp. 1243-1256.

[Hua 2005] Huang W., Beevers K. Topological Map Merging. The International Journal on Robotics Research. – Sage science press, 2005, 24. vol. 8, pp. 601-613.

[iRo 2012] iRobot. Internets: http://www.irobot.com/us/robots/home/roomba.aspx, aplūkots: 16.08.2012.

[Ish 1993] Ishioka K., Hiraki K., Anzai Y. Coorperative Map Generation by Heterogeneous Autonomous Mobile Robots. IJCAI’93 Workshop on Dynamically Interacting Robots, august 1993. – Chambery, 1993, pp. 57-67.

[Ko 2003] Ko J. et. al. A practical decision-theoretic approach to multi-robot mapping and exploration. IEEE/RSJ International Conference on Intelligent Robots and Systems, October 2003. – Nevada, 2003, pp. 3232-3238.

[Kon 2003] Konolige K. et. al. Map merging for distributed robot navigation. IEEE/RSJ International Conference on Intelligent Robots and Systems, October 2003. – Nevada, 2003, pp. 212-217.

[Lak 2005] Lakaemper R., Latecki L. J., Wolter D. Incremental multi-robot mapping. IEEE/RSJ International Conference on Intelligent Robots and Systems, august 2005. – Edmonton, 2005, pp. 3846-3851.

[Mar 2005] Martinelli, A.; Pont, F.; Siegwart, R., "Multi-Robot Localization Using Relative Observations," Robotics and Automation, 2005. ICRA 2005. Proceedings of the 2005 IEEE International Conference on , vol., no., pp.2797,2802, 18-22 April 2005

[Nik 2013] Nikitenko A., Liekna A., Ekmanis M., Kulikovskis G., Andersone I. Single Robot Localisation Approach for Indoor Robotic Systems through Integration of Odometry and Artificial Landmarks. Scientific Journal of Riga Technical University, Computer Sciences, 2013

[Par 2000] Parker L. E. Current State of the Art in Distributed Autonomous Mobile Robotics. 5th International Symposium on Distributed Autonomous Robotic Systems, October 2000. – Knoxville: Springer, 2000, pp. 3-12.

[Rou 2002] S. Roumeliotis and G. Bekey, "Distributed multirobot localization," *Robotics and Automation, IEEE*, vol. 18, no. 5, pp. 781–795, 2002.

[Roy 2000] Roy N., Dudek G. Collaborative Robot Exploration and Rendezvous: Algorithms, Performance Bounds and Observations. Autonomous Robots. – US: Springer, 2000, 11. vol., pp. 117-136.

[RTU 2013] Par projektu „Daudzaģentu robotizētas intelektuālas sistēmas tehnoloģijas izstrāde", Internets: http://www.rtu.lv/content/view/5226/1867/lang,lv/, pēdējoreiz aplūkots 01.09.2013.

[Sim 2000] Simmons R. et. al. Coordination for Multi-Robot Exploration and Mapping. AAAI National Conference on Artificial Intelligence, 2000

[Thr 1998] S. Thrun and W. Burgard, A. probabilistic approach to concurrent mapping and localization for mobile robots. Autonomous Robots, 25, pp. 1–25, 1998

[Thr 2001] Thrun S. A Probabilistic Online Mapping Algorithm for Teams of Mobile Robots. International Journal of Robotics Research. - Sage science press, 2001, 20. vol., pp. 335-363.

[Thr 2002] Thrun S. Robotic Mapping: A Survey. Exploring Artificial Intelligence in the New Millennium. – Morgan Kaufmann, 2002, pp. 1-35.

[Thr 2005] Thrun S. Probabilistic robotics. The MIT Press, 2005

[Top 2010] Topal S., Erkmen E., Erkmen A. M.: A Novel Map Merging Methodology for Multi-Robot Systems. In Proceedings of the World Congress on Engineering and Computer Science, Vol I, 2010, pp. 383-387.

[Wil 2002] Williams S. B., Dissanayake G., Durrant-Whyte H. Towards Multi-Vehicle Simultaneous Localisation and Mapping. IEEE International Conference on Robotics and Automation, May 2002. – Washington: IEEE, 2002, pp. 27-43.