Institute for Design and Control of Mechatronical Systems

Diploma Thesis:

Groupbased exploration and mapping with sensoric under-equipped robots

Proof of concept

Author: Wolfgang Wögerer

Supervisors: Prof. Dr. Luigi del Re

Thomas Passenbrunner

Finished: June, 2014

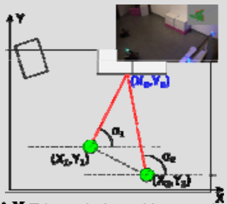
Abstract

Most approaches to cooperation in the field of cooperative robotic are based on dividing the tasks between different robots to decrease the needed time. Still, one robot could do the work on it´s own, as it has all the needed functionality implemented. Inspired by the function of an ant-hill the concept of robots who are not able to work alone was tested with specially desiged robots and scaled up to the use of dozends within a simulator. This simulator had to get as close to the real life robots as possible to get valid data for statistic analysis. In this thesis three robots where built to work together and give real life information to test and refine the simulated environment. The scalability and statistical analysis was done with a large amout of simulated robots as the construction and testing would have been to much effort.The basis of the cooperation was mapping of an unknown environment with triangulation. The robots scan their surroundings by marking a spot with a laser pointer by one robot and scanning the angle of this point in relation to all the other robots. This forces at least two robots to work together find one point of the environment. The results of this thesis show that the approach is valid and distributed functionality on many robots has many advantages like smaller, cheaper robot configurations, system stability in case of robot failure and specialized robots for certain tasks.

Triangulation and Grid maps

In this project every point of the environment was scanned by using laser lights and cameras to find the needed angles α1 and α2. If the position of the robots is known, the position of the lightpoint can be calculated. This gives two types of information:

- At the position of the lightpoint is an obstacle
- Between the robots and the lightpoint is no obstacle.

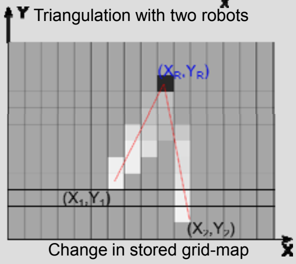


Triangulation with two robots

If n robots are used to scan one point the amount of triangles can be calculated with:

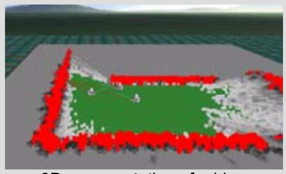
$$N_t = \sum_{1}^{n-1} i$$

Due to measurement errors none of these points will be on the exact same position. So the quality of the triangles will be estimated by the size of the smallest angle within one triangle. A small angle leads to bigger position errors if the measured angles are not correct. With this quality-indicator the impact of one estimation on the resulting map will correleate with it´s accuracy.



Change in stored grid-map

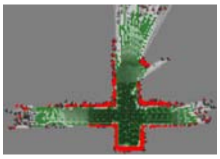
A very effective way to store map information is by creating a grid and filling it with information about the individual cells. Every cell has a value between 1 and 0 to store if there is a obstacle inside this cell or not. If a point within a cell the „blocked“ value will increase according to the estimation accuracy. The free cells between the robots and the point get their „blocked“ value decreased. As the accuracy of the pointposition is better if the points are closer a non linear function is used to decrease the value of the closer cells more.


3D representation of gridmap

Top level logic

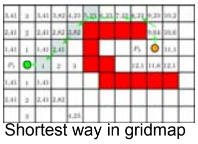
Create cluster of cooperative robots

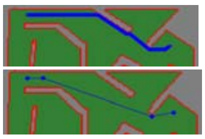
Possible positions on the map can be evaluated for their usability to gain new information about the environment. This process estimates how much a robot could learn if it would be on this particular spot. The more unknown fields around a position, the higher the maximum gain. Then the robots are distributed in the interesting areas in groups. The size of the group is calculated with a penalty for groupsize (to small is bad, to big is bad) and the distance a robot has to travel to join the specific group. The last step is correction of the positions within one group to prevent blockage of the line of sight if all the robots stand behind each other.


Maximum gain on map

Repositioning

To move the robots around in the free area, the shortest way to move from one spot to another is needed. Calculating the distance of each cell to the starting point is done by jumping from cell to cell. Only shorter ways are stored resulting in the shortest way through the grid. Still the grid forces horizontal, vertical and diagonal movement which is not very efficient. Therefore all the corner points are checked if they are really needed and get removed if the direct connection of the neighbor points is possible. This is done until no more point can be removed.


Shortest way in gridmap


Optimizing way

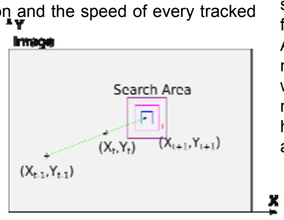
Simulations

Part of this project was a simulator for statistical analysis of bigger robot groups with different properties. With the real robots to verify and improve the accuracy of the simulation, it represents a important tool to try many scenarios in a short time. This has shown that the accuracy of the measured angles has a big influence on the speed how fast the robots explore the map, but the resulting map is still very good, This proves that a bigger amount of robots will compensate for bad sensors. As accurate sensors are very expensive this opens up the ability to use many cheap robots instead of a few expensive ones.

Optical sensors

Find Light points

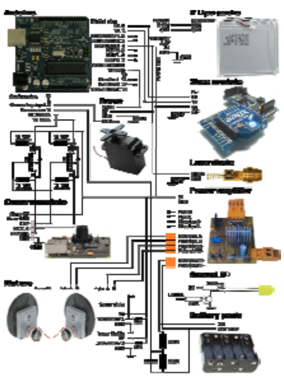
The only sensor to explore the environment is the camera. All the needed information are light points either a laser pointer dot or LED lights. To find the points on the images from the sensor every one is processed in three steps. First the shutter time is reduced to create a very dark image where only bright lights are still visible. Now the points are found and tracked in two seperate ways. First the full image processor scans the whole image using a grid with a random row and column count. Every cell is scanned for one light point and the resulting points are handled by the point tracker. Processing the whole image takes to much time for a high frame rate so this is only done every 15 frames. In the mean time the point tracker scans only the part of the image where the point should be. This is estimated with the information where the point was in the last two images to find the position and the speed of every tracked point. This is much faster and enables point tracking with a frame rate over 18 frames per second. The found points are sent from the camera-module to the robots microcontroller to triangulate positions or for calibration purposes.



Hardware

Creating the Robot

At the time this project started, only a few experimental robots where available. So the robots had to be created from scratch. Only the camera module, the microcontroller board and the xbee module where preassembled. All the other boards and connectors where desiged and built specifically for this project. This includes a connector board, a power amplification module, bus-voltage converters and all the needed power supply stabilisation and noise filtering to power the robots. At the end of this project many experimental robots where available on the market so nowadays the hardware would not be such a big challange anymore.



Conclusions and Outlook

In this project we checked the usability of sensoric under-equipped robots in groups. We were able to design methods to organize and coordinate the robots in a simulated environment and refine the result with real robots in experiments. It turned out that we had to do this in two steps. The first was to create the methods and test them on simulated robots, which where created parallel to the real ones. This had the advantage of increased reliability of the robots and a faster test cycle time to do many simulations in a short time. Additionally the amount of robots is not limited in the simulation. The second step was to confirm with the real robot, that the simulated robots behave like the real ones and correct the simulations until they do.

With these realistic simulated robots we were able to run many simulations with hundreds of different settings and compare the performance and reliability of the tested algorithms. We found that the approach with these dependent robots does not only work, it is very stable against loss of single robots, as the whole group behaves like a single organism. Due to the clustering the problem is broken down to smaller parts that are solved by individual groups, so the solution scales very good with the amount of robots.

The next step would be the use of specialized robots sent along with the other ones to carry sensors for heat or toxins without the need to equip them with the basic functionality like navigation sensors, because the other robots do this for the whole group. This would create a completely new form of flexible and dynamic robot functions where groups could switch specialized robots when they need it, eliminating the need to reequip a single robot for different tasks with different sensors.