Multi-robot System for Vacuum Cleaning Domain

Agris Nikitenko, Janis Grundspenkis, Aleksis Liekna, Martins Ekmanis, Guntis Kulikovskis, and Ilze Andersone

Riga Technical University,
Faculty of Computer Science and Information Technology,
Department of Systems Theory and Design,
1 Kalku Street, Riga, LV 1658, Latvia
{agris.nikitenko,janis.grundspenkis,aleksis.liekna,
martins.ekmanis,guntis.kulikovskis,ilze.andersone}@rtu.lv

Abstract. This demonstration paper presents a prototype multi-robot system of vacuum cleaning robots. System was designed with the aim to join multiple robots in a team able to accomplish tasks that are beyond the capabilities of a single robot.

Keywords: Multi-robot systems, vacuum cleaning.

1 Introduction

Vacuum-cleaning robots are becoming more popular among household users. Such robots are capable of cleaning individual rooms and small offices. Nevertheless, cleaning large indoor areas such as hangars is still a challenge, which had not been addressed by commercial applications yet.. To address this challenge we have developed a multi-robot vacuum-cleaning system. We use existing iRobot Roomba [1] vacuum cleaning robots as robot platforms. We improved them with additional capabilities required for participating in a multi-robot system by attaching a custom-made extension on top of the robot. This extension includes a computational element with an Intel Atom[2] processor, a web-camera for localization purposes, additional bumper sensor and an additional battery. The added computational element provides means to address localization, map building and path planning as well as interaction functionality with team members.

System architecture is partially centralized, where a central server plays the central role by allocating tasks to individual robots and merging individually created robot maps. When a task is assigned to a robot, the task is executed autonomously and the server has no direct control over the low-level actions of the robots.

System operates in the following way. User initiates the task allocation procedure by requesting a cleaning operation using the user interface. This request is then sent to server and it spreads the whole cleaning area into smaller sub-regions called responsibility areas. Each robot is assigned a responsibility area. After receiving such

[1] http://www.irobot.com
[2] http://www.intel.com/content/www/us/en/processors/atom/atom-processor.html

Y. Demazeau et al. (Eds.): PAAMS 2014, LNAI 8473, pp. 363–366, 2014.

an assignment, robot travels to the responsibility area and performs the cleaning operation. After the operation is complete, next unassigned responsibility area is assigned to that robot. The process continues until all areas are properly cleaned.

2 Main Purpose

The main purpose of the developed solution is to join multiple existing robots in a multi-robot system to accomplish tasks that are beyond capabilities of a single robot. Particular task can be defined as area coverage or sweeping task where multiple robots are required to cover a particular area, however we focus on the overall system architecture and implementation not the sweeping algorithms themselves. The following constraints are assumed to be held: a) capabilities of a single robot are not enough to cover the whole area; b) each robot is capable of autonomously covering some part of the area; c) a task can be decomposed into subtasks in such a way that each subtask can be accomplished by a single robot. An example of such a task is cleaning a large area warehouse using vacuum-cleaning robots that are designed to clean a hotel room – no robot is capable of cleaning the whole warehouse in a reasonable time, but the warehouse can be divided into smaller areas each of which can be cleaned by a single robot.

We use iRobot Roomba vacuum cleaners as robot platforms. These robots already have a built-in algorithm for cleaning a designated area. Our objective was to use these robots as a basis and extend their capabilities to produce a multi-robot system. To reach the set objective the following interrelated challenges were addressed:

- *Task Allocation*. To use the built-in cleaning algorithm of the robots, they must each operate in a distinct designated area. This was achieved by splitting the total cleaning area into smaller ones called responsibility areas which are then assigned to individual robots.
- *Localization*. For the robot to be able to travel to assigned responsibility area and perform cleaning afterwards, it is essential for the robot to "know" at least the relative coordinates of itself and the target area. To provide the necessary means of localization, we use a combination of artificial landmarks (glyphs) observable by robot onboard camera and robot odometry (wheel encoders) to estimate robot position and angular direction in the environment. Our approach is based on assumption that the indoor environment is available for landmark use – it means that it is possible to install specially designed landmarks for robot positioning in global coordinates. Details on applied localization techniques are published in [1] and [2].
- *Path Planning*. To travel safely while avoiding obstacles, an obstacle-free plan for the path towards the goal area must be constructed. We use Rapidly-Exploring Random Tree (RRT) algorithm [3] for robot path planning. The aim is to find the shortest path from current position to destination point while avoiding obstacles in the way. A modified version of original RRT algorithm is used as published in [4].
- *Map Building*. To construct an obstacle free-plan, it is essential to know the coordinates of the obstacles, as well as the source and destination positions. This is why a map of the environment is required. Robots build their map using only the factory-installed iRobot Roomba sensors. We do not add any additional sensors motivated by sustaining reasonable cost of the solution. More specifically we use short-range infrared and bumper sensors to detect obstacles in the environment. We start with an

empty map of the environment with (approximately) predefined size. Map is populated with information about obstacles and "free spaces" while robots move in the environment. Refer to [5] for detailed information on applied mapping algorithm.

- *System Architecture.* Organization and management of multi-robot system must be supported by appropriate software architecture. To address this problem a partially decentralized solution is adapted in the following way. An application runs onboard each robot and is responsible for the low-level control of the robot and serves as individual localization and mapping source. A server component is also introduced and serves as a central point of the system. Detailed description of system architecture can be found in [1].
- *Robot Hardware Enhancements.* To cope with challenges previously described in this list, a specialized hardware platform is designed and developed. This platform is attached on top of iRobot Roomba vacuum-cleaning robots. It consists of a computational element with Intel Atom processor, a rechargeable battery to support the computational element, a web camera for glyph recognition used in localization and an additional bumper sensor to compensate extra height of the platform. Only a few screws and a UART cable are necessary to connect the platform to an existing robot – no interference with internal robot design is required.

3 Demonstration

Demonstration consists of 8 minutes long video showing the system in operation, complimented by the interviews of main contributors (see Fig. 1). The interview include complementary information about the used methods, approaches and achieved results as well as outlines the future plans of the team.

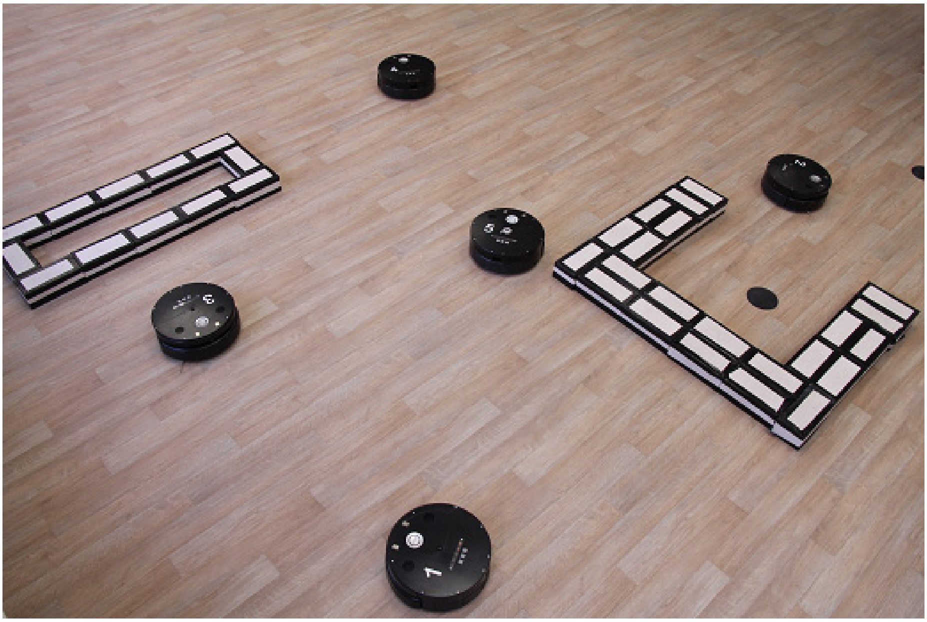

Fig. 1. System operation

4 Conclusions

By solving the challenges of task allocation, localization, path planning, map building, system architecture and robot hardware enhancements, a working multi-robot system is produced, successfully joining multiple robots in a unified system capable of completing tasks that are beyond the capabilities of a single robot.

Acknowledgements. This work has been partly supported by ERDF European Regional Development Fund project 2010/0258/2DP/2.1.1.1.0/10/APIA/VIAA/005 Development of Intelligent Multiagent Robotics System Technology.

References

1. Liekna, A., Nikitenko, A.: Architecture and. NET Implementation of Multi-Robot Management System. Applied Computer Systems 14, 59–66 (2013)
2. 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. Applied Computer Systems 14, 50–58 (2013)
3. LaValle, S.M., Kuffner Jr., J.J.: Rapidly-exploring random trees: Progress and prospects. In: Proceedings Workshop on the Algorithmic Foundations of Robotics (2000)
4. Nikitenko, A., Ekmanis, M., Liekna, A.: RRTs Postprocessing for Uncertain Environments. In: Proceedings of the 2013 International Conference on Systems, Control and Informatics (SCI 2013), pp. 171–179 (2013)
5. Andersone, I., Liekna, A., Ņikitenko, A.: Mapping Implementation for Multi-robot System with Glyph Localisation. Applied Computer Systems 14, 67–72 (2013)