Technical issues of MRL Virtual Robots Team RoboCup Asia Pacific 2018, Kish Island – Iran

Mohammad H. Shayesteh[1], Mohammad M. Raeisi[1]

[1] Islamic Azad University of Qazvin, Electrical, IT & Computer Sciences Department , Mechatronics Research Laboratory, Qazvin, Iran

{m.h.shayesteh, mahdrsi}@gmail.com

Abstract. In this paper we describe MRL Virtual team preparation to take part in RoboCup Asia Pacific 2018. Regarding new RoboCup 2018 challenges, we tried to design a new software for exploring virtual rescue robots based on ROS framework [1]. We present our new software in many different parts such as multi-robot manual control, SLAM and autonomous exploration.

1 Introduction

In the virtual robot competition a disaster environment is simulated which could be explored with a team of rescue robots. It based on a simulator and presented under an open source package, a high fidelity simulator on the Gazebo simulator [2]. Within this simulator research teams can setup multiple agents whose capabilities closely mirror those of real robots. This simulator currently features wheeled as well as some sensors and actuators. Moreover, teams can easily develop models of new robotic platforms, sensors and test environments.

MRL Virtual Robot have participated since 2006 in various RoboCup completions such as: IranOpen, Kharazmi and WorldCup. Our major focus is on developing four Wheels and Areal robots. We have been champion on 2013 and 2014 WorldCup competitions and our base research area is on: Autonomous systems, SLAM and Multi Agents systems. MRL team consist of M.Sc. and BC.s students in different fields such as Artificial Intelligent, Software Engineering and Information Technology Engineering. Most of mentioned researches area are defined as thesis’s topics. Mechatronic Research Laboratory is depend on Islamic Azad University of Qazvin.

2 Team Members

The team members and their contributions are as follows:

- Mohammad H. Shayesteh: GUI[1], SLAM[2], Map Merge
- Mohammad M. Raeisi: Navigation, Autonomous Exploration

3 New Software

With the latest changes in Virtual Robot league, all teams are using “ROS Framework” to create better modules for SLAM, Navigation, etc. to develop a suitable program to manage multi robots. So due to these changes we prepared to design a new software to be a dependable program for use ROS capabilities and add other modules in the future.

This new system is designed with C++ language programming and QT platform for multi-robot driving application. The software is consist of multi-robot setup, control, visualization and camera viewer. In this TDP we show the most important abilities and sections of our software follows as:

Setup Environment:
This section provides a wizard form to spawn how many robots for each rounds and you can configure robot name, init positions or robot topics and etc.

Multi-Robot Control:
A software with suitable GUI to control many robots and switch between manual or autonomous driving mode. In this section, users can select how many robot to be spawn by setup form and this dashboard considers a dynamic view based on system configuration.

Visualizer:
A dock panel in main software for visualize the robot trajectories, explored maps and marking the victims. Explored maps is prepared from our map merged package and operator can follow real time map in competition rounds.

Camera Viewer:
A separate section that shows RGB and Thermal cameras in multi-window. Operator can easily monitor all of robot cameras in a single form. In this window we are design a widget for each robot that shows RGB and Thermal camera in a right way.

---

[1] **G**raphical **U**ser **I**nterface
[2] **S**imultaneous **l**ocalization **a**nd **m**apping

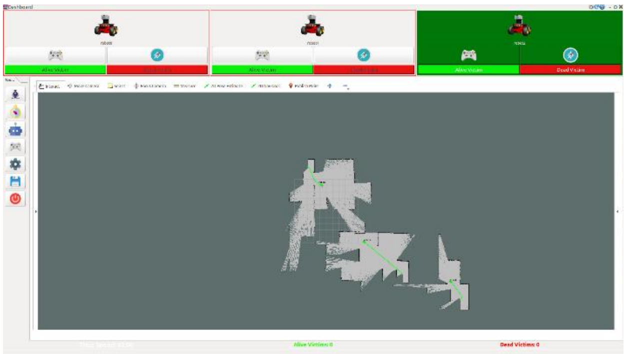

Fig 1. Multi-robots control dashboard and dock visualizer

Fig 2. Multi-robots camera monitoring window

As shown in Fig 1, you can do some necessary tasks like opening cameras window, switching all robots to manual drive or autonomous exploration and saving explored map from left panel. In top panel you change robot status between manual and autonomous, mark dead/live victims on map separately and also explored map with robot trajectories is shown in button dock panel.
As shown in Fig 2, all robot RGB/Thermal cameras shows vertically/horizontally next to each other and operator can easily monitor all robot windows in a single form.

4 System Overview

Software in ROS is organized in packages. A package might contain ROS nodes, a ROS-independent library, configuration files, a third-party piece of software, or anything else that logically constitutes a useful module. The goal of these packages it to provide this useful functionality in an easy-to-consume manner so that software can be easily reused.

In our new software, each challenges such as SLAM, Navigation, Exploration and multi-robot control is separated from each other by standard ROS packages. We customizied and enhanced many ROS package and also design some new other packages as our requirements.

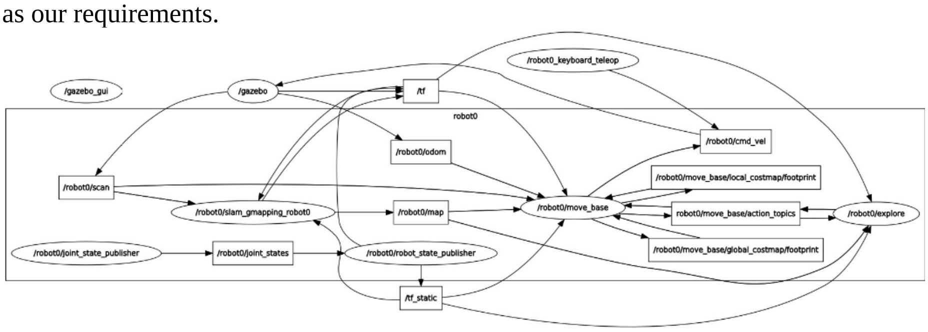

Fig 3. Robot node with its stacks for different challenges

As shown in Fig 3, each robots acts as a node with their SLAM, Navigation and Exploration stacks and they controls with main software that described in section 3.

5 SLAM & Navigation

Scan matching as a basic part of SLAM has a key role in localization and even Mapping of mobile robots. In our previous researches, we implemented ICEG [3] as Scan matching method and Grid Mapping in previous competitions.

This year for SLAM and Navigation challenge, we uses ROS Packages. They are standard in implementation and have a good performance in real time rounds. For mapping, The gmapping package[4] provides laser-based SLAM (Simultaneous Localization and Mapping), as a ROS node called slam_gmapping and the algorithm is used with a specific packages enhanced for our team, which using a 2D Hokoyou type laser scanner which embedded on all of our Pioneer-3at Robots.

In addition, each robot have a local slam node and they use this service in many others sections like navigation and exploration. Finally we merge all of local maps for every robot with multirobot_map_merge package [5]. This package is customized based on our requirements and able publish global positions of robots and also their trajectories.

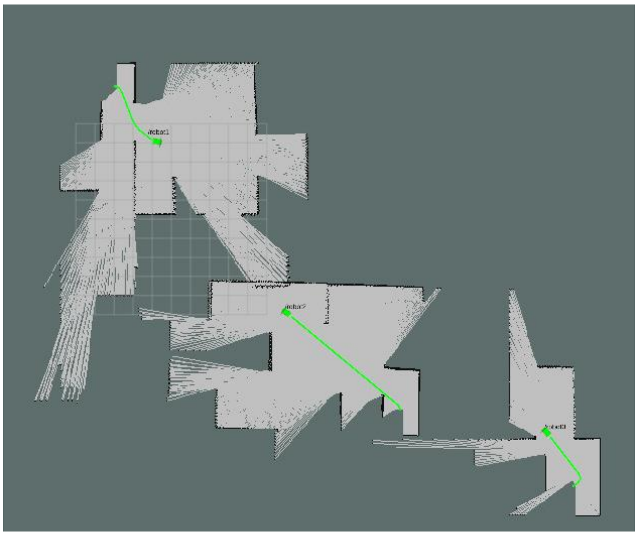

Fig 4. Merged maps of spawned robot

As shown in Fig 4, each robot sends its local map to map merge package, after that all local maps merge together then publishes by a specific topic for visualization and another purpose. Also robot trajectories and positions is calculated in our customized map merge package and draw in visualizer panel.

For more precise explain core of our 2D-Navigation System, It is based on base_local planner which we uses Teb local planner [6, 7, 8, 9, 10] algorithm that provides clear path from published maps and optimizes the robot's trajectory with respect to trajectory execution time, separation from obstacles and compliance with kino-dynamic constraints at runtime.

6 Autonomous Exploration

One of the main purposes in virtual robot league is autonomous exploration. Each robot should automatically explore the unknown areas based on definite rules.

For this goal, we are using explore_lite [11] package and this package provides greedy frontier-based exploration. When node is running, robot will greedily explore its environment until no frontiers could be found. Movement commands will be send to Navigation section.

7 Conclusion

In this paper we are designed our new software based on ROS framework which are needed for autonomous systems. On the other hand, we tried to design an autonomous exploration for wheeled robots to search in the disaster environments. Our future task is to design a Multi-Robot Exploration system to navigating a group of robots parallels based on ROS framework. This part helping us to make better decision in autonomous rounds and explore wide area in indoor environments.

References

1. Quigley M., Gerkey B., Conley K., Faust J., Foote T., Leibs J., Berger E., Wheeler R., Andrew N. : “ROS: an open-source Robot Operating System”, “ICRA Workshop on Open Source Software”
2. https://github.com/reyhanehpahlevan/RoboCup2018RVRL_Demo
3. Taleghani, S., Sharbafi, M. A., Esmaeili, E., Haghighat, A. T., 2010, ICE matching, a robust mobile robot localization with application to SLAM. IEEE 22nd International Conference on Tools with Artificial Intelligence (ICTAI), Arras, France, Oct 27–29.
4. http://wiki.ros.org/gmapping
5. http://wiki.ros.org/multirobot_map_merge
6. C. Rösmann, F. Hoffmann and T. Bertram: Integrated online trajectory planning and optimization in distinctive topologies, Robotics and Autonomous Systems, Vol. 88, 2017, pp. 142–153.
7. C. Rösmann, W. Feiten, T. Wösch, F. Hoffmann and T. Bertram: Trajectory modification considering dynamic constraints of autonomous robots. Proc. 7th German Conference on Robotics, Germany, Munich, May 2012, pp 74–79.
8. C. Rösmann, W. Feiten, T. Wösch, F. Hoffmann and T. Bertram: Efficient trajectory optimization using a sparse model. Proc. IEEE European Conference on Mobile Robots, Spain, Barcelona, Sept. 2013, pp. 138–143.
9. C. Rösmann, F. Hoffmann and T. Bertram: Planning of Multiple Robot Trajectories in Distinctive Topologies, Proc. IEEE European Conference on Mobile Robots, UK, Lincoln, Sept. 2015.
10. C. Rösmann, F. Hoffmann and T. Bertram: Kinodynamic Trajectory Optimization and Control for Car-Like Robots, IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Vancouver, BC, Canada, Sept. 2017.
11. https://github.com/hrnr/m-explore