Robocup 2010 – Virtual Robot League

Edris Esmaeili aliabadi[1], Sanaz Taleghani[1] ,Mohammad Shirzadi[1]

[1] Islamic Azad University of Qazvin, Electrical Engineering and Computer Science Department, Mechatronics Research Lab,
Qazvin, Iran
{esmaeili, taleghani ,shirzadi}@mrl.ir

Abstract. This paper describes the main features of the MRL Virtual Robots team which intends to participate in Singapore 2010 competitions. Virtual Robots is an environments that a combination of state of the art algorithms of Robotics and Artificial Intelligence fields are needed to deal with its challenges. In the following we describe our approach for the main challenges such as Simultaneous Localization and Mapping (SLAM), Exploration, Navigation, Cooperation and etc.

Keywords: Localization, Mapping, Exploration, Navigation.

1 Introduction

Nowadays Robotics and Artificial Intelligence are the center of attention of many researchers. USARSim provides us with an environment in which the conjunction of these two fields occurs. In this environment, a disaster (usually an earthquake) is being simulated in indoor and outdoor scenarios. The goal is to gather a map of a previously unknown environment which would provide information about the situation, victims, damages and etc. To overcome the goal, a combination of the state of the art algorithms of different fields needs to be implemented. These fields include to Localization, Mapping, Machine Vision and Image Processing, Robot Navigation, Robot Communication, Robot Coordination, Modeling and etc. In this paper, we describe our approaches to these different challenges which we would use to participate in Singapore 2010 Virtual Robots Competitions.

Our team members and their contributions in team are:

- Edris Esmaeili : Coordinator and Software Developer
- Sanaz Taleghani : Localization, Mapping (SLAM)
- Mohammad Shirzadi : Software Developer, Exploration
- Amir Panah : SLAM

2 Localization and mapping

2.1 Localization

SLAM is a fundamental problem in mobile robot system which has been studied extensively in the robotics literature. SLAM is the process that an autonomous robot moves in an unknown environment and then computes position from its perception and gradually builds a map of this environment.

So It is crucial for robot to be able to find an accurate position . We have tested many different localization algorithms like Dp-SLAM, FastSLAM, CoreSLAM which were used particle or kalman filter family. These approaches might require a very large number of particles or landmarks, this limitation strongly may cause the worst-complexity of run time. Furthermore In these feature-based approaches the computational complexity will increase with growing of environment size and number of landmarks.

One good way among other methods is to match range scans, made by, e.g. a laser range finder, taken at different locations at different times in the environment and to update the position estimate according to the match result. Such algorithms are well known as scan-matching algorithms. According to the match result, the translation and rotation (dx, dy, $d\theta$) are used to update the robot position.

During our researches about different localization algorithms, we found out that most of the current SLAM methods based on Scan Matching have different bottleneck, therefore we've innovated our own SLAM algorithm named GSLAM.

Like so many other scan matching method, GSLAM works with features (landmarks) of a map and performs simple geometrical rotation and translation searches to match the features of the current observation of time step t+1 and the features of the previous observations of time step 1 to t. Consequently, for the localization matter, the negative of these rotation and translation changes of features are applied to the robot's state.

GSLAM has its own Feature map which does not provide us with a high quality and comprehensive map. Therefore, we use Grid mapping to generate our final desired map of the environment.

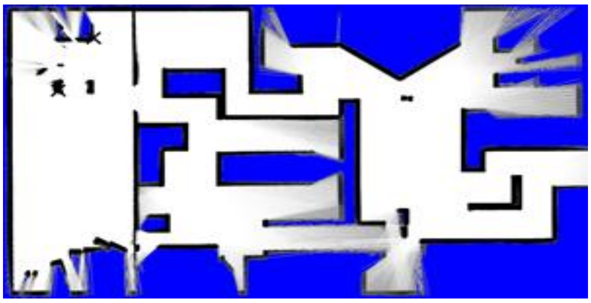

Fig. 1. A Generated map using GSLAM and Grid Mapping

This algorithm has a problem, when feature is insufficient it would fail. Therefore we intend to use other algorithms such as Extended Kalman filter-Which was applied on INS data- to cover this problem. We used a new prediction model that is based on GSLAM Result and Sensor Data such as INS or encoder.

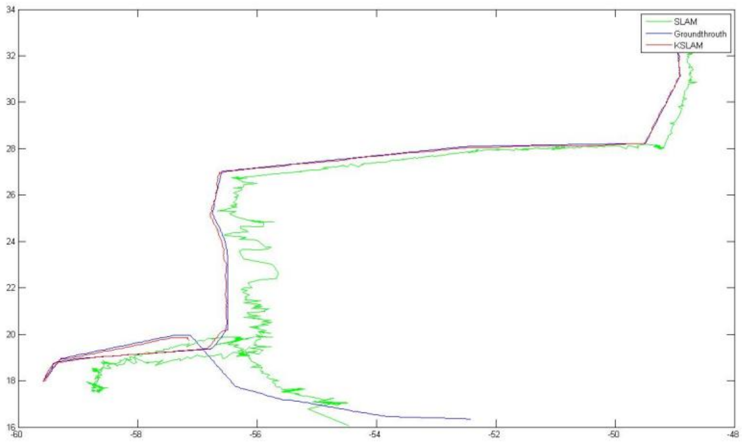

Fig. 2 Real path and GSLAM output and EKF&GSLAM output

2.2 Mapping

While moving through an environment the robot is required to derive a map from its perception. We've used to use a Grid Mapping method which was improved by ourselves. This algorithm received the sensor data from each robot in a real time way that had a large overhead on the center. But it had an advantage: After the disconnection of robots from center we didn't lose the map so far till that time. We are currently using a new algorithm which contains both advantages that is based on line produced by feature extraction methods which are sent to center and we create the map based on these lines.

3 Exploration And Navigation

3.1 Exploration

Previous year we improved our Fuzzy Algorithm by changing the system from center base to distributed agents, in this way, we shared process of exploration between robots and reduced a lot of process from center (Communication Station). Using Fuzzy Algorithm with new parameters such as recognition of corridors, rooms and the widest area for searching, help us to make a better decision for managing robots.

This year we added new decision making system for our robots using graph theory. Each robot contains a graph that represents information about other robots such as positions and signal strength of other robots, partial exploration map. In this graph each vertex is equal to a robot and each edge between two vertexes is equals to signal strength of related robots. Robots will send their information to all other robots which are in the same area, thus all robots in same area have others information.

In addition we've used flooding algorithm to broadcast information from a robot to other robots that are out of range. In this algorithm each robot will sent other robots position and signal strength between other robots and itself to other robots that are in range, when robots received this information, they will merge it with their information center and then they send it to others that are in their range.

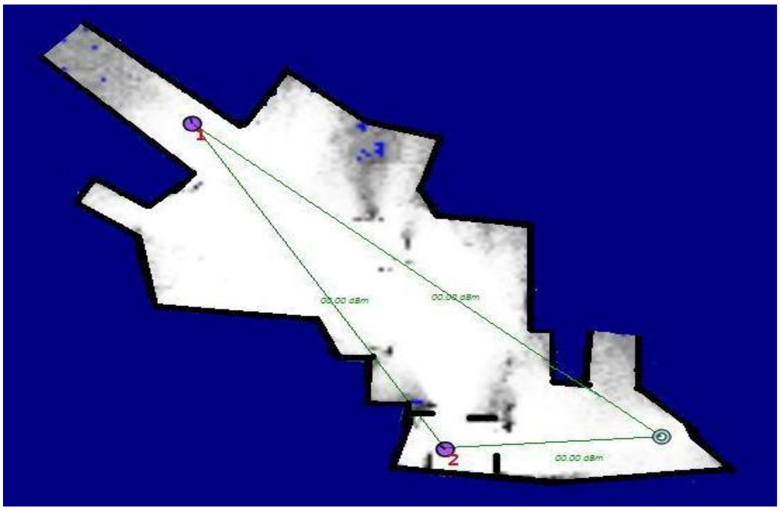

Fig. 3. Each robot has a graph including positions and signal strength of other robots, partial exploration map

Also we added a time stamp to avoid replacing older information with newer while merging it into information center. In this system all robots information are updated and robots will used this information for their decision making.

Faster decision making, faster updating information between robots and less network traffic are benefits of this system. Also if each robot loses its connection to the center, it can route its message using other robots.

3.2 Navigation

Navigation is also the science of getting mobile robots from place to place, a problem that is central to mobile robotics and has been subject to extensive research. This involves answering the three questions “Where am I”, “Where am I going?” and “How do I get there?

We’ve used a combination of A* and Potential Field to find the optimum way for reaching to the target location. Using A* wasted a lot of time in decision making and also wasn’t applicable in unknown environments, in addition to it wasn’t necessary for the robots to remain still during the decision making process which resulted in less exploration area.

Thus we’ve decided to implement an electromagnetic environment based on our strategy that assigned homonymous poles to robots and obstacles and non-homonymous poles to the leader and the target. Then the speed of each wheel is calculated from the resultant of these vectors and it causes the robots to be in motion continuously. We developed a program to test this algorithm that would simulate this environment.

4 Graphical User Interface

Last year we’ve developed a GUI to represent robots activity and enable operator to manage robots for their mission. In this system we used GDI+ technology for drawing each element of simulation environment such as robots and their mission, victims, etc.

Low speed in drawing and taking high cpu usage was problems of our GUI. Also we had concurrency problem because there was one resource that robots and GUI could write and read on it respectively, and times of reading from resource was more than write on it.

These problems make us to think about new methods for resource sharing between robots and other readers such as GUI. Main idea in our system came from database snapshot theory. We defined a main resource in communication station for writing all information and named it Data Center and a snapshot of this resource for others that want read information. We will take new snapshot from Data Center when times of writing is minimum.

The Number of threads for writing information in Data Center was many, because of having many threads in queue waiting for writing information in Data Center, in this way, we separate each type of information in Data Center and made specific channel for each type to write. Also we added different queue to store different types of information. This method will help us to find information faster to represent. History of Data Center will save on hard disk to consider at the end of simulation.

As mentioned above, using GDI+ took high CPU usage and was slow in drawing. For this reason we changed our system from GDI+ to DirectX. Also we used WPF instead of windows application and made better management on our application.

For making management and developing of our software as simple as possible, we developed a plug-in based system. Stability, Simplicity and designing different part of system in different application and joint them to make complete software are benefit of this system.

In our application, the definition of plug-in is each part of system that could be isolated from system and developed in different application. We defined a single rule for plug-ins. Each plug-in must follow this rule and developed in this structure. Plug-ins that is out of this structure will refuse to load in main application. The main benefit of plug-in based systems is adding new features to the system without changing in the main application.

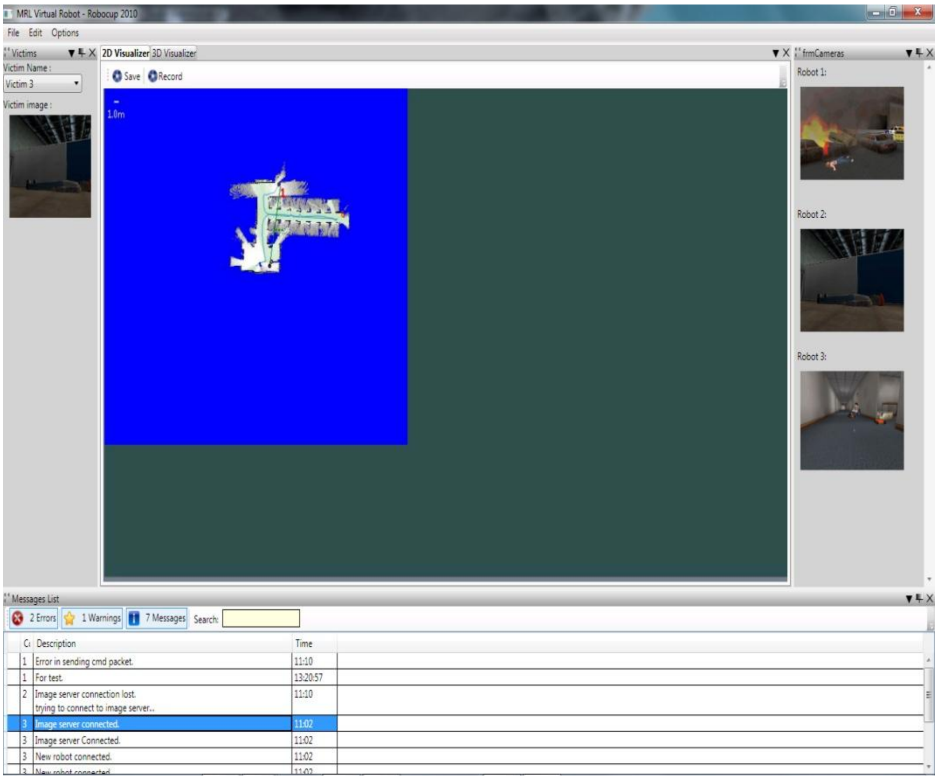

Fig. 4. Snapshot of GUI

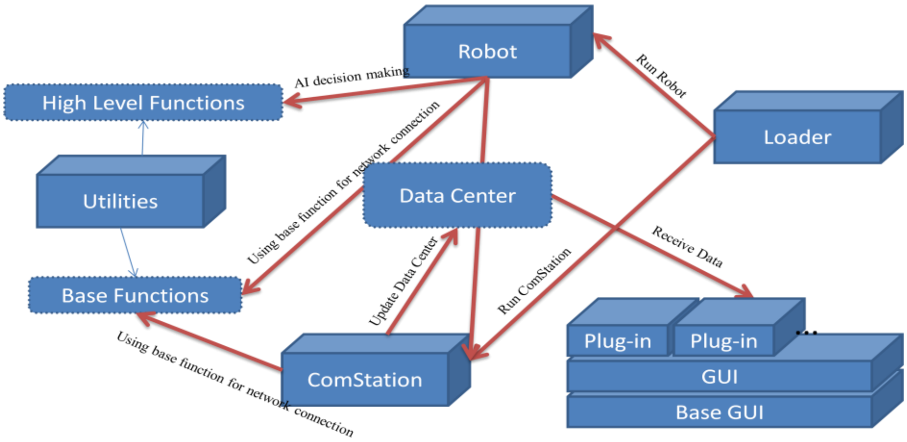

Fig. 5. MRL Data Flow Diagram

5 Cooperation

For optimizing the cooperation of robots we tried that each robot performs according to the location of other robots and also forming and developing the area of a graph. A robot's own new location is chosen based on this graph and other parameters that was mentioned in section 3-1. In this part optimization algorithms such as Genetic were used.

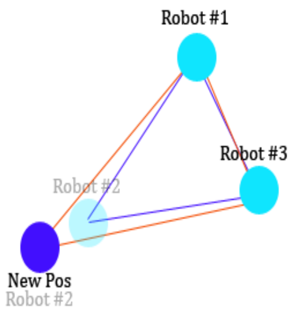

Fig. 6. Generated Graph

5.1 Communication

We use simple messaging protocols for communication between robots and operator. WSS is utilized to deliver our messages to their intended target. If a robot is not accessible to receive operator's command, communication station will send command using other robots that are accessible. Also robots that can't send their data to communication station, they will send it to other robots that are in range, in this way, communication station can receive data from all robots that are in environment. Also we implement such algorithm to find minimum distance from a robot to communication station.

5.2 Coordination

For coordinating the robots in decision making in Distributed model we decided to broadcast the locations of the robots to them in some particular frequent intervals that the robots would realize the presence of other robots around them in the electromagnetic field so they won't face any trouble in finding their direction.

6 Conclusion

Our 3.5 years experience of working in Virtual Robots field lead us to an innovation about the SLAM problem and some other different method gradually. These methods are still in progress, but the promising results which we got from these methods encourage us to continue our work.

References

1. Phung, S. L., Chai, D., Bouzerdoum, A.: A Universal and Robust Human Skin Color Model Using Neural Networks. In Proceedings of INNS-IEEE Int'l Joint Conf. Neural Networks, vol. 4, pp. 2844-2849, July (2001).
2. Phung, S. L., Chai, D., Bouzerdoum, A.: Skin segmentation using color pixel classification: analysis and comparison. IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 27, no. 1, (2005).
3. Bailey, T., Durrant-Whyte, H.: Simultaneous Localisation and Mapping (SLAM): Part I - State of the Art. Robotics and Automation Magazine, September (2006).
4. Bailey, T., Durrant-Whyte, H.: Simultaneous Localisation and Mapping (SLAM): Part II - State of the Art. Robotics and Automation Magazine, September (2006).
5. Montesano, L., Minguez, J., Montano, L.: Lessons Learned in Integration for Sensor-Based Robot Navigation Systems. International Journal of Advanced Robotic Systems, vol 3, no. 1, pp. 85-91, (2006).
6. Minguez, J., Montesano, L., Lamiraux, F.: Metric-Based Iterative Closest Point Scan Matching for Sensor Displacement Estimation. IEEE Transactions on Robotics. Vol. 22, pp. 1047- 1054.

7. Zhao, L.: Dressed human modeling, detection, and parts localization. Ph.D. thesis, University of Carnegie mellon, Pittsburgh, USA, (2001).
8. Felzenszwalb, P. F., Huttenlocher, D. P.: Efficient graph-based image segmentation. International Journal of Computer Vision, vol. 59, no. 2, pp.167-181, (2004).
9. Hazon, N., Mieli, F., Kaminka, G. A.: Towards robust on-line multi-robot coverage. In Proceedings IEEE International Conference on Robotics and Automation, (2006).
10. Kong, C. S., Peng, N. A., Rekleitis, I.: Distributed coverage with multi-robot system. In Proceedings of IEEE International Conference on Robotics and Automation, (2006).
11. 1. Eliazar, A. I., Parr., R.: DP-SLAM 2.0. In: IEEE International Conference on Robotics Automation, New Orleans (2004).
12. Eliazar, A.I., Parr, R.: Hierarchical Linear/Constant Time SLAM using Particle Filters for Dense Maps. In: Advances in Neural Information Processing Systems 19, MIT Press (2005).
13. Lisien, B., Morales, D., Silver, D., Kantor, G., Rekleities, I., Choset, H.: Hierarchical Simultaneous Localization and Mapping. In Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'03), Las Vegas (2003)