Research and Implementation of Cooperative Control for ROS Mobile Robot

Saijie Zhang, Huailin Zhao*

School of Electrical and Electronic Engineering, Shanghai Institute of Technology, Shanghai, 201418, China

*E-mail: *zhao_huailin@yahoo.com*
www.sit.edu.cn

Abstract

In this paper, the control of multiple mobile robots TurtleBot is studied for the robot operating system (ROS). A distributed control and communication platform is built to solve the problem of data transmission between multiple robots, and the configuration of robot workstations that can be monitored and remotely controlled is completed. On this basis, the SLAM mapping and autonomous navigation of the robot are completed. The SLAM mapping algorithm of Gampping is selected. With the help of Gazebo simulation platform and Rviz visualization tool, the local path planning of TEB is used to realize the autonomous navigation and multi-point navigation of the robot based on the known environment. On the basis of the constructed distributed control system and the control of a single robot, the research on the coordination of multiple robots is further carried out, and the synchronous control of the robot is realized by using a single node and multi-thread method. By changing the commands issued by the topic, the control of multiple robots for different attributes is released synchronously to achieve more accurate and more synchronous robot motion control. With the help of TF tools, multi-machine following and multi-machine formation are completed, and based on the path planning of a single robot, the navigation of multiple robots is realized.

Keywords: ROS, Multi-robot cooperation, Distributed control

1. Introduction

Robot Operating System (ROS) is an open source standard for the robotics industry. It integrates sensors, motors, and controllers into reusable modules through a distributed messaging architecture to achieve mobility, operation, navigation, and recognition tasks [1]. With the continuous development of the Robot Operating System (ROS), ROS has gradually entered various fields to complete various task requirements. However, when dealing with some complex tasks, it still seems to be somewhat inadequate, and sometimes it is necessary to cooperate with the staff to meet the requirements. Therefore, the research on multi-robot cooperative control is particularly important. The realization of multi-robot collaboration can not only cope with complex work scenarios and complete complex work requirements, but also improve work efficiency and enhance the intelligence of robots. Moreover, with the development of artificial intelligence, robots have ushered in new development opportunities. Based on the ROS multi-robot patrol algorithm simulation and benchmark test framework, the development of multi-robot systems, virtual reality and robot interaction has become popular. In the future, ROS will develop rapidly in the fields of navigation and positioning, 3D object recognition, motion planning, multi-joint manipulator motion control, machine learning, human-machine interaction, robot collaboration and so on.

2. Cooperative control of ROS mobile robot

The basis of collaboration between robots is two-way communication, because in the process of collaboration, task allocation, data transmission and information generalization are involved. Robots need to communicate frequently to obtain location information and working status of other devices. Therefore, to realize the cooperative control of mobile robots based on ROS, we first need to build a communication platform, and all devices are connected to the platform, and use this as a benchmark for communication and wireless connection. At the same time, in order to facilitate the management and monitoring of all robot devices, a remote workstation device needs to be set up. The workstation can remotely connect any robot device for management, and can also monitor the working status and task progress of all robots.

2.1. *Design of distributed control system*

A distributed communication and control platform is built for robots to provide network support for robot collaboration.

A. *Static IP settings and master-slave IP configuration*

In order to ensure the normal communication function between robots and between robots and remote control terminals (workstations), each device needs to be connected to the same network, which can be provided by the wireless signal of the router or maintained by the hot spot service of the mobile phone. Because the general wireless network will assign different IP addresses to different users, although different robot devices are in the same network, because the IP address is different, and can not communicate, so the network provider is required to manually set up static IP and set up a multi-machine communication environment in the. bashrc file of each device. The setting of static IP is very simple. It only needs to change dynamic acquisition to manual acquisition in IPV4, and set fixed IP address, default routing, subnet mask and DNS.

ROS is a distributed computing environment. A running ROS system can contain multiple nodes distributed on multiple computers. According to the configuration of the robot operating system, multiple nodes between multiple computers, different nodes running on the same computer may need to communicate with each other at any time. Therefore, ROS requires a complete two-way connection between all robots on all ports for network configuration, and each computer must announce itself through a name that all other computers can parse.

B. *ssh Remote Connection*

In the distributed architecture of multi-processors, file transmission may be frequently involved between different robot systems. For example, ROS programs are written on the workstation, and eventually need to run on the client. It is necessary to upload related directories and files from the workstation to the client. In fact, the design of the control system of multiple robots often adopts the mode of remote management of the workstation. After the network and function of each robot client are configured for the first time, in order to facilitate the update of the robot firmware, the processing when the operation is wrong and the iterative upgrade of the function package, it is also convenient to monitor the whole system. A remote workstation solves these problems. The problem is how to realize remote connection on the workstation and log in to the client for management. ROS gives the answer, which provides two remote control tools, ssh and vnc. Here only introduce ssh this way.

SSH (Secure Shell) is an encrypted network protocol for secure network services such as secure data communication, remote command line login, and remote command execution between two networked computers. It encrypts all data transmitted between two computers and provides secure and authenticated communication over an insecure network. It is commonly used by system administrators to manage remote servers and by developers to securely access and transfer files between machines. The whole process of data transmission is transparent and safe, using openssh tools will enhance your system security. The SSH implementation architecture is divided into two parts: client and server. The client is the requester and requests to upload files. The server is the receiver, receiving and downloading files from the client. The workstation belongs to the client and sends data to the robot. The robot belongs to the server. The implementation process is divided into four steps: install SSH client and server respectively ; the server starts the SSH service ; client remote login server ; realize data transmission.

C. *Optimization of ssh usage*

The client needs to enter the password every time it logs on to the server, and it also needs to enter the password when uploading and downloading files, which actually brings great inconvenience to the transmission of files. How to optimize the use of SSH, make the server trust the client, give the client a permission without password access, simplify the login steps with the help of the key, realize the secret-free login, and improve the efficiency of the operation. The implementation idea of optimization is to generate a pair of public key and private key. The private key is stored locally, and the public key is uploaded to the server. When each login, the local directly uploads the private key to the server. The server can find the matching public key, and it is identified as a legitimate user. The ssh connection is directly created, and the client does not need to enter the password.

D. *Communication between robots*

The ROS kernel (roscore) is the basis of ROS operation, which has a parameter server. A running ROS has and only has one ROS kernel, and everything on ROS depends on this kernel [2]. ROS is a distributed framework based on the development and integration of many Nodes (nodes). Therefore, how to connect and manage many nodes is a problem that ROS must solve, which introduces a communication mechanism. In the communication mechanism, ROS Master (master node, node manager) manages and schedules the communication process between nodes in the network, and provides a service for configuring global parameters in the network. There are three basic communication modes of ROS, namely topic communication (publish-subscribe mode), service communication (request-response mode) and parameter server (parameter sharing mode).

2.2. *Synchronization control of mobile robot*

There are two control schemes to realize the synchronous control of robots. The first scheme is that the controller synchronously publishes multiple topics, and each topic corresponds to the ns attribute of different robots. The detailed process is shown in Fig. 1.

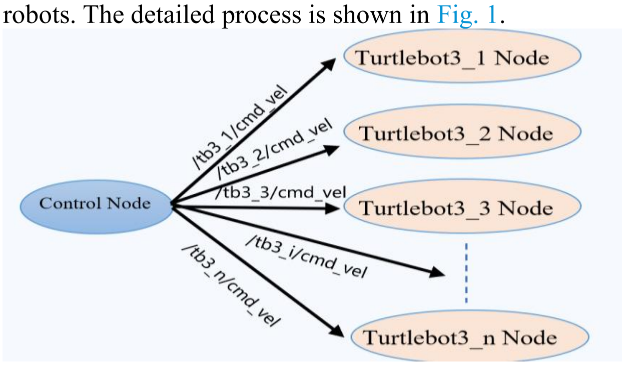


Fig. 1 The controller synchronously publishes multiple topics

The second scheme is to create an intermediate node, which receives the control instructions of the controller and processes them. After adding the ns attributes of different robots, it is released to the robot for control. Fig. 2 shows the implementation of scheme 2.

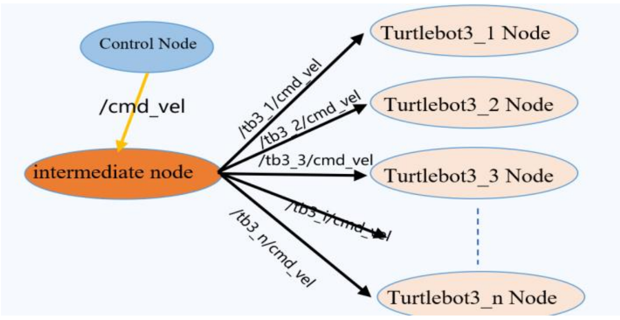


Fig. 2 Intermediate nodes assist in publishing

Both schemes allow the robot to obtain the same motion control. Scheme 1 needs to modify the code of the original given controller. After modification, it is impossible to control a single robot independently. Scheme 2 retains the original control node and still retains the control characteristics of a single robot. When multiple robots need to be controlled, only one more intermediate node needs to be run. Therefore, Scheme 2 is more practical.

2.3. *Mobile robot queuing to follow*

In order to realize the following of multiple robots, the following between two robots is considered first. If robot A follows robot B, the relationship between the position and attitude of the two robots must be known. Then through the coordinate transformation, the coordinate system transformation matrix between the two robots can be obtained, as well as the displacement vector randomly transformed to the pilot. The robot follows the simulation as shown in Fig. 3.

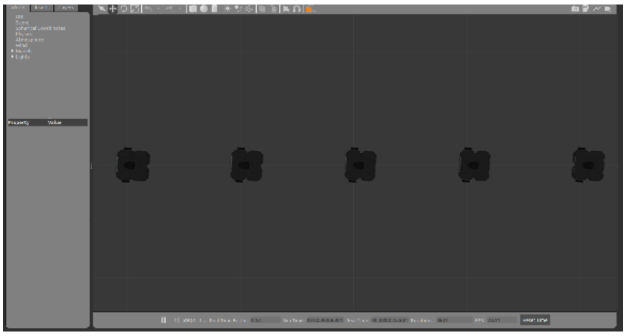


Fig. 3 Multi-robot following simulation

As long as the relationship between the coordinates is calculated in real time, ROS provides four methods to obtain the coordinate relationship of robot pose, which can be obtained by SLAM mapping, odometer, global camera, motion capture device and so on. After obtaining the coordinates, it can be sent to the TF tool in ROS by broadcast. Then subscribe to the relationship between the two robots in the TF tool. The distance and attitude angle difference between the two rob real-time follow can be achieved through the TF tool. ots can be obtained by the pose relationship between the two robots. The speed and angular velocity instructions are issued by PID control to realize the following control of the robot.

The key to realize multi-machine following is to use TF (Transformations Frames) tool, which is a tool for monitoring and calculating robot pose information, providing pose monitoring, coordinate transformation and other functions. In the robot operating system, the TF toolkit includes Broadcaster, Listener, and TF conversion tools. The broadcaster broadcasts the position relationship between each robot and the world coordinate system to the TF tool through the topic; listener obtains the relative position of the two robot coordinate systems by viewing the TF toolkit.

2.4. *Multi-robot formation of mobile robot*

The principle of formation and following is that the TF tool is needed to obtain the relationship between the robot coordinate systems, but the following needs the transformation of the coordinate systems of the pilot and the random. The random moving target point is the coordinate origin of the pilot. The pilot moves, and the coordinate origin moves accordingly. TF will broadcast its pose in real time to control the random speed and angular velocity to achieve the purpose of real-time following. The coordinate system and coordinate transformation are shown in Fig. 4.

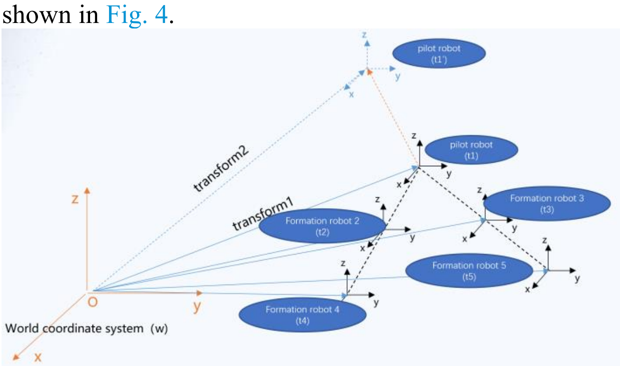


Fig. 4 Formation coordinate system and coordinate transformation

Compared with the following, the realization of the formation needs to establish a virtual coordinate system of the specified position around the leader, and obtain the change of the position and attitude of the virtual coordinate system following the leader coordinate system in real time through the TF tool. The remaining formation aircraft track the position and attitude of the specified virtual coordinate system respectively, and the robot formation movement of various formations can be realized. By using the TF tool and the basic communication method to publish the virtual coordinate system of each specified position, the robot formation problem is transformed into the tracking problem of the target point. It is necessary to modify the callback function in the control of the pilot robot to realize the release of the specified virtual coordinates. Because the virtual coordinate is relative to the position of the pilot robot, it can be adjusted according to the required formation. In this paper, three formation modes are demonstrated, including horizontal line, triangle and square escort formation, and the position and transformation of virtual coordinates of triangle formation are given. They are shown in Fig. 5, 6 and 7.

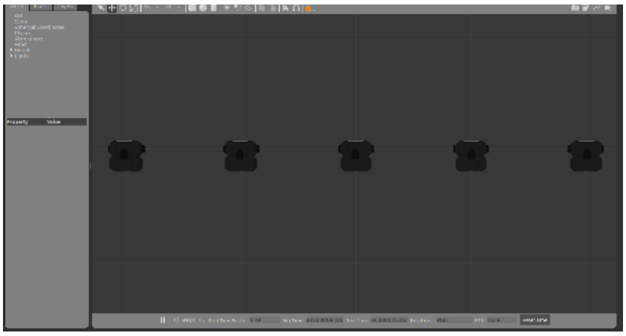


Fig. 5. Horizontal formation simulation

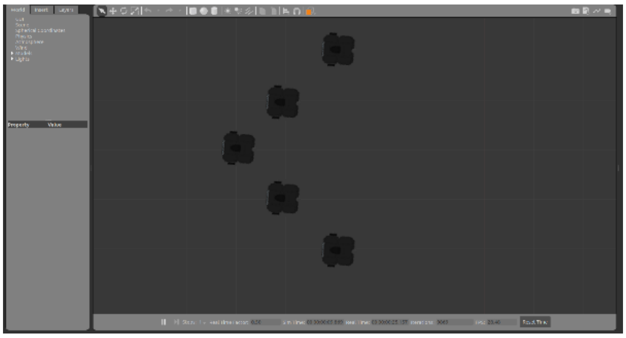


Fig. 6 Triangle formation simulation

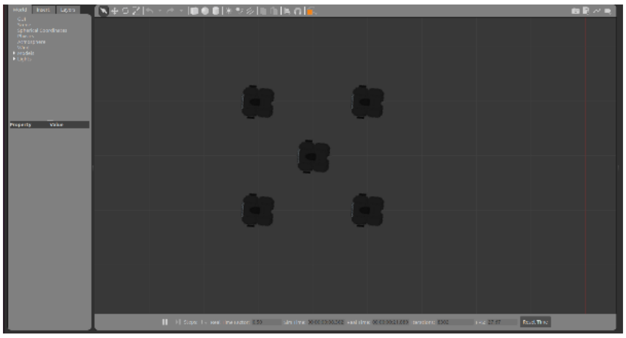


Fig. 7 Square escort formation simulation

2.5. *Cooperative navigation of mobile robot*

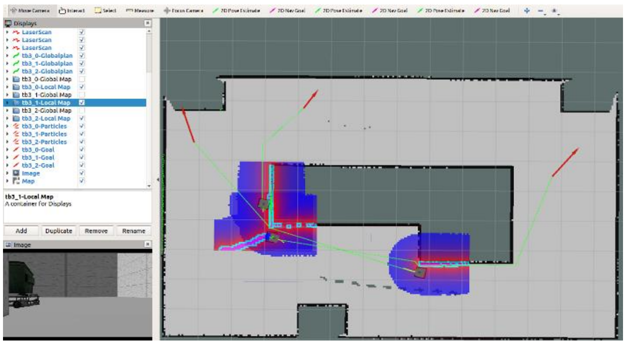


Fig. 8 Robot cooperative navigation

Fig. 8 shows the scene of three robots navigating at the same time in the same environment, and the three robots will determine their own movement routes through cooperation to avoid the risk of collision with each other, and move to the established target point in the shortest time and most efficiently.

A. *Robot path planning*

Before path planning, it is necessary to build a grid map of the robot 's working environment through SLAM mapping [3], because the robot 's navigation process needs to create target points on the map, and in the path planning process, the path planned by the robot will be displayed on the map in real time. Mapping algorithms are also diverse, including unscented Kalman filter UKF-SLAM, FAST-SLAM, Gmapping, etc [4], [5]. In this paper, the Gmapping mapping algorithm based on lidar data is selected. The basic idea of Gmapping algorithm is to obtain the information of the surrounding environment through lidar when the robot moves, and then construct a map and locate the position of the robot through filtering and matching algorithms.

In this paper, the TEB algorithm [6] is used to avoid obstacles. Firstly, several rough paths that can reach the target point are taken as the original paths. Then, various constraints are considered on each path, such as time, collision, speed, acceleration, etc., and these constraints are used to construct the optimization problem. Finally, the optimization problem of each initial path is solved, and a better path than initialization is obtained. Then, a better path is selected from the optimized path, and the result of local path planning can be obtained.

B. *Robot cooperative navigation*

ROS multi-robot cooperative navigation is to complete the cooperative work of multiple robots in the same environment to achieve navigation and exploration of the environment. In ROS, multi-robot cooperative navigation usually uses multiple nodes to achieve. Each robot has its own navigation nodes, which can exchange information with each other through ROS communication mechanism. In cooperative navigation, robots need to understand each other's position and state in order to coordinate their actions. When implementing ROS multi-robot cooperative navigation, the following aspects need to be considered:

Communication mechanism: ROS provides a variety of communication mechanisms, including Topic, Service and Parameter Server. In multi-robot cooperative navigation, topics and services are usually used to achieve communication between robots.

Position estimation: The robot needs to accurately estimate its position and posture. Laser radar, vision sensors and other sensors can be used to achieve position estimation.

Obstacle avoidance algorithm: In the navigation process, the robot needs to avoid obstacles and avoid collisions.

Cooperative planning algorithm: In multi-robot cooperative navigation, robots need to coordinate their actions to avoid mutual interference. Collaborative planning algorithms can be used to achieve collaborative action.

3. Conclusion

Since the birth of ROS in 2007, the research and development of this distributed framework system with nodes as processing units and functional units has flourished, and robot distributed control has great prospects in the development of service-oriented robots and warehouse logistics robots. As one of the popular development directions of robots in the future, multi-machine collaboration not only provides a solution to deal with complex working environments in the working process of robots, but also greatly improves the working efficiency of robots. The working requirements of service-oriented robots are not only the quality of service, but also the efficiency of their work. With the fast-paced operation of society, we are mostly in a ' fast-food ' attitude towards life. The efficiency of a single robot is limited, and the management of multiple robots is complex. Therefore, a multi-machine cooperative control system that can be autonomously managed, information shared, divided and autonomous, and autonomously coordinated is developed, that is, swarm intelligence. It meets the requirements of high efficiency and simple management. Warehouse logistics robot is a system that

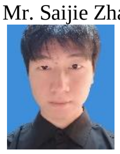
pursues time efficiency. How to ensure its work efficiency and achieve the required accuracy at the same time is more convenient for the management of multiple robots in the whole system. Multi-machine coordination of distributed control may give its answer. Let the robot coordinate autonomously, and at the same time share data in real time and allocate tasks, which is convenient for management and efficient.

References

1. Vincenzo DiLuoffo, William R Michalson & Berk Sunar. (2018).Robot Operating System 2. International Journal of Advanced Robotic Systems (3). doi:10.1177/1729881418770011.
2. Yahui Qi, Xiuzhen Wu, Qijie Chen, Shi Yan & Peng Zhang. (2022). Implementation of multi-UAV cooperative control based on ROS networking. (eds.) Proceedings of the Fifth National Conference on Cluster Intelligence and Cooperative Control in 2021 (pp.104-107).
3. Chengkan Huang. (2020). vision-based laser SLAM AGV initial positioning and path planning research (master 's thesis, Zhejiang University of Technology). https://kns.cnki.net/KCMS/detail/detail.aspx?dbname=CMFD202101&filename=1020439617.nh
4. Xuebing Wang, Qingsheng Guo, Yong Wang, Qizhi Liu& Zhiwei Wei. (2019). Feature extraction and automatic recognition methods for map images. Surveying and mapping and spatial geographic information (09), 28-32.
5. José Carlos Rangel,Miguel Cazorla,Ismael García-Varea,Cristina Romero-González & Jesus Martínez-Gómez.(2019). Automatic semantic maps generation from lexical annotations. *Autonomous Robots* (3). doi:10.1007/s10514-018-9723-8.
6. C. Roesmann, W. Feiten, T. Woesch, F. Hoffmann and T. Bertram, "Trajectory modification considering dynamic constraints of autonomous robots," ROBOTIK 2012; 7th German Conference on Robotics, Munich, Germany, 2012, pp. 1-6.

Authors' Introduction

Mr. Saijie Zhang



He received the B.S. degree from Shanghai Institute of Technology, Shanghai, China. His main research interests is automation.

Dr. Huailin Zhao

He received his PhD from Oita University, Japan in 2008. He is a professor in the School of Electrical & Electronic Engineering, Shanghai Institute of Technology, China. His main research interests are robotics, multi-agent system and artificial intelligence. He is the members of both IEEE and Sigma Xi.