AN EXPERIMENT IN REALIZING AUTONOMOUS AND COOPERATIVE BEHAVIOURS BETWEEN MULTIPLE MOBILE ROBOTS

Suparerk PREMVUTI and Shin'ichi YUTA
Institute of Information Science and Electronics
University of Tsukuba
Tsukuba Ibaraki, 305 JAPAN

Abstract: Research into autonomous mobile robots has been actively pursued around the world. In the first stage of development, researchers wanted to realize a robot that could move and work autonomously. Introducing mobile robot for applicationsin factories, one robot cannot achieve all the tasks required. Several robots working cooperatively is essential in a factory. This paper describes an architecture for autonomous mobile robots which is extended to support cooperative behaviour between other mobile robots. The implementation and experimentation on real robots is presented.

1. Introduction

Research in the field of autonomous mobile robot is aimed toward the goal of creating robots with high autonomy. Once the target is reached, research will focus on building multiple robots which cooperate with each other.

First, the concept of each robot in the multiple robots sytem must be investigated. The authors propose a concept of 'modest cooperation' to allow the sharing of resources; for example, sharing the work environment, work tools. This sharing of resources is done without centralized control.

To make the robot cooperate with each other, a communication network module which is responsible for communicating between robots is required, and software that supports programing of multiple robots is designed.

An experiment with three autonomous mobile robots was performed to proof the effectiveness of the concept, and the results are reported in this paper.

2. The principles of respect of others and modest cooperation

Considering the cooperation of autonomous mobile robots which each having its own objective such that the robots do not disturb one another can be thought as fundamental requirement of autonomy as well as cooperation.

Actions that can be considered as disturbances among robots are:
(1) Standing in the ways of others.
(2) Interfering in operations of other robots by using active sensors that emit something like light that causes sensors of the others not to function properly.
(3) Using some work tools, that another robot wants to use.

All these disturbances are problems concerning the exclusive use of spatial resources. For instance, roadways are necessary resources for robots to perform their work. At a crossroad, at one time, only one robot can cross. And for active sensors, they must emit a scanning light or other kinds of radiation waves exclusively.

The disturbances among robots occurs when there is competing for the same spatial resources. Preventing a disturbance can be achieved by each robot giving the right of exclusive use of a spatial resource to other robots. By this way, a competition for resources between robots can be avoided.

The authors suggest that actions by each robot *should be achieved without any discussion among the robots. Rather, a robot should let the other robots use the resources autonomously. Each robot should recognize that competition will occur if it tries to access the resource.* The authors call this type of cooperation, 'The Modest Cooperation'. The modest cooperation can be thought as a fundamental function of a self-contained autonomous mobile robots.

One of the main features of modest cooperation is that it is based on 'local spatial information'. A *robot is concerned with the use of spatial only when* other robots are near by. Then, under modest cooperation, while each robot is working on its own task, the robot must examine its surroundings and respond to the situation properly. A concrete algorithm of the modest cooperation must be designed while keeping the following items in mind.
(1) When all robots in the system perform modest cooperation based on the algorithm, competition of resource access must not occur.
(2) When all robots act according to the algorithm, a deadlock should be minimised.
(3) When some robots don't act according to the algorithm, there should not be any great risks of competition for resources.

3. Deadlock and its solution

When several robots give way to each other *modestly, in some situations, a deadlock can occur.* So, when implementing the modest cooperation in the system, it is necessary to design an algorithm that minimises the chances of deadlock. In reality, functions of a robot cannot be executed as exactly as planned. Some problems can occur, for instance, delays in communicating data or erroneous sensor data. Although a system structure and its algorithm are designed by providing a sufficient margin, it is very difficult to find an algorithm that provides no deadlock.

When a deadlock occurs, the parties concerned have to discuss the way to solve the problem by using the following prescription.
(1) Each robot must recognize a deadlock and broadcast the recognization to all the other robots.
(2) By discussing over the communication network, the concerned robots and the resources will be identified.
(3) A solution to the deadlock problem is a common objective among the concerned robots and is solved by the robot that is assigned to a leader position.
(4) All other robots move under instructions of the leader to resolve the deadlock.
(5) Then, each robot begins to execute its original mission.

4. Necessary hardware to the modest cooperation

When a robot acts according to modest cooperation, it must know where the other robots are and what they are doing. For this reason, an environment *sensor and perception are necessary.*

The role of sensors is to examine whether there is any possibility of collision with other robots, while using the spatial resource. It is difficult for currently available sensors to sense the positions of other robots. On the other hand, each robot knows the information concerning its own movements, for example, its current position and the next action. It is better to prepare a communication network among robots, which broadcast information about the current positions of each robot to the other robotsconstantly. By this method, each robot can receive information about other robots, as if it were using its own sensor to sense autonomously. The robot uses this broadcasted information instead of sensory information for deciding its future movements.

Thus, in order to realize cooperation among robots, besides mechanisms necessary for independent autonomous motions, a robot needs powerful sensors and a communication mechanisms.

5. An example of cooperation in a multi robot system

The best way to illustrate the concepts discussed in this paper is by example. Firstly, a basic problem to solve was to allow several robots to move in the same space without colliding with each other using the cooperation concepts.

The problem:

[An objective for each robot]

Robots, which posses ultrasonic range sensors and an inter-robot communication network system, move in the same environment of routes network. Each robot is provided with an individual planned path to execute. These paths are able to be specified freely. So, it is possible that some parts of paths may run over or cross each other. The width of routes in the environment is large enough for only one robot to move along, and cross roads are limited to three branches.

[Deadlock description]

A deadlock can occur when a couple of robots want to move in the same path section. Then, a shunting algorithm must be used to solve the deadlock problem

Shunting can be thought as a common goal for the robots caught in the deadlock. One of robots involving with the deadlock is selected as a leader. This robot finds a solution and sends instructions to the other robots.

6. Experimentation: Implementation of cooperative behaviour in a multi robot system

The authors have implemented the cooperative behaviour on the YAMABICO family of autonomous mobile robots [1].

6.1 About the robots and their environments

The control system of each 'Yamabico' robot consists of several functional modules (refer to fig.1). Each module is responsible for a specific function in the robot. The modules are as follows:

(1) Master module

The module executes a program for the robot. While executing the program, the module will pass functions called in the program to a specified module.

(2) Locomotion module

The module executes locomotion functions sent from the master module. This module ensures the robot follows the specified course.

(3) Ultrasonic range finder module

The module measures the distance between the robot and obstacles in four direction; front, back, left, right.

(4) Communication network module [2]

The module supports broadcasting communication and one to one communication. Broadcasting communication is used for periodically sending information about the current state of the robot, for example, the current position of the robot. One to one communication is used between a leader and a slave while shunting in order to solve a deadlock.

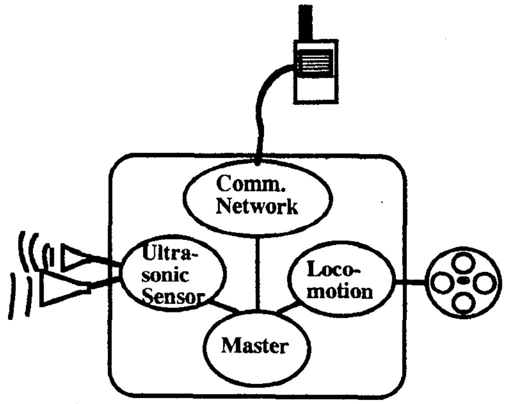

Fig.1 Multi module structure of a robot

The Yamabico has ultrasonic sensors and uses wheels with driven by servo DC motors for locomotion. The motor axis is attached to an encoder which counts turns of the motor. This count is used to estimate the current position. The ultrasonic sensors can detect the distance between the robot and object to the front, back, left and right. Thus, they can be used for making the robot run along the roadways.

The environment for robots is straight paths which formed a network. The width of each path is 90 centimetres with both sides bounded by sound reflectors. Only three branches cross sections are allowed in the path network. A map of the environment is given to the robots prior to operation. While the robots are moving along the paths, unpredicted obstacles such as pedestrians may be in front of robots at anytime.

6.2 System configuration

Yamabico robots are equipped with a communications network(refer to fig.1). The purpose of this system is to support cooperative actions between multiple autonomous mobile robots. The system has no central processor for controlling the communication network, or for controlling the behaviour of the robots. Each robot is an independent unit. Thus, this robotic system can be considered to be highly autonomous.

6.3 Software for realizing cooperative behaviour

On the master module of each Yamabico, besides system processes, there are processes for controlling cooperative behaviours. Important processes are the 'behaviour control process' and 'resource management process'(refer to fig.2).

6.3.1 How to manage resources

For a robot to perform its task in a 'modest and cooperative' way, it must know about the occupation state of the shared resources. Then, an information panel which provides this kind of information is used. An information panel is a memory area which is used for communication between processes by writing and reading to the memory. While moving, each robot broadcasts information about its status to all other robots. At the same time, each robot receives information broadcasts from the others and puts this information into the information panel.

Management of the information panel is done by a process called the 'resource management process'.

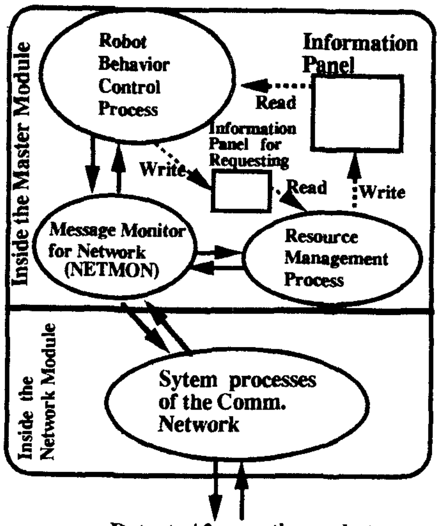

Fig.2 Show main processes concerning cooperative behavior and their data flows

The main tasks of this process are updating information about resources which are sent from other robots and try to secure resources requested by the 'behaviour control process'. This process will be explained later.

The result of a request by the behaviour control process is shown on a similar informational panel. In short, the resource management process is responsible for updating and managing the informational panel and the behaviour control process uses information in the informational panel.

6.3.2 The behaviour control process

The main task of this process is to observe the environment surrounding the robot and then deciding what has to be done next. Basic movements of the robot, for example, running along a path, stopping before a collision occurs, are controlled by this process. Moreover, the process is responsible for realizing higher level decision making, for example, sharing roadways with other robots, the shunting procedures to solve a deadlock at a crossroad.

The behaviour control process is written in ROBOL/0 which is a robot language developed by the other members of the authors' research laboratory [3]. The main functions of the process can be devided into 4 parts.

(a) Route runner

'Route runner' is an algorithm which allows a robot move along a given path.

The process sends proper locomotive commands to the locomotion subsystem to make the robot move. While the robot is moving, the process monitors the estimated position by dead reckoning using the locomotion subsystem and distances to sidewall of the path measuring by the ultrasonic sensor.

When a robot reaches an estimated positon for correcting its trajectory, it will read the distance between the robot and sidewall and compare it with a precalculated distance. If there is any difference between the two distance, the process will calculate a new estimated position and call a function to set the internal position of locomotion subsystem to the new position.

After the locomotion subsystem gets the correct position from the behaviour control process, it automatically adjusts movements of the robot to maintain the correct trajectory.

Another task of the process is to avoid collisions with obstacles in front of the robot. When a distance from the front ultrasonic sensor is less than a specified value the process will call a function to stop the robot.

(b) Sharing resource by using principle of modest cooperation

The path for a robot consists of several unit straight roads which were called 'sections' connected in a row. The section is considered to be a kind of resource for movements of a robot. Before a robot enters the next section, it must secure the section by request the section from the resource management process. The section can be secured for use if and only if the resource is not occupied by other robots.

The behaviour control process requests the section by writing the request on an information panel. Then the process checks the status of the required section on the information panel and decides whether the robot will enter the section or not. The robot will enter the next section only when the information panel show that the section is already secured.

After the robot enters the next section, the behaviour control process will try to secure the next section it expects to enter. In the case that the next section cannot be secured, the robot will stop at the crossroad.

(c) Deadlock detection

While stopping at a crossroad the behaviour control process will check the information panel to see if the next section is avaliable or not, and decide if a deadlock state has occurred. The process will examine the number of robots which have stopped at the crossroad and the directions that each robots want to go. From this information the process can determine whether a deadlock has occurred or not.

(d) Solving the deadlock problem

When the behaviour control process detects a deadlock, the process will try to gain leadership status. The leadership status can be considered as a resource. The behaviour control process must request from the resource management process the leadership status in the same way as requesting for the next section of road to run. The result of the request, be it a leader or be it a slave, is shown on the information panels. The robot whose behaviour control process gains the leadership status will generate a shunting plan that controls the movement of one or two slave robots. After shunting is finished, the leader robot will declare through broadcasting that its leardership is over. The robots at the crossroad are once again in the same status and can continue moving autonomously again.

6.4 The experiment

The environment for three autonomous 'YAMABICO' mobile robots is shown in fig.3. Each robot has been assigned with a course to move along repeatedly. Some parts of the courses must be shared between some robots.

After the three robots start moving, they travelled along assigned course successfully(Photo 1). While running, the robots shared the roads by waiting at the crossroads. Sometimes a deadlock state occurred, but the concerned robots successfully solved the problem by shunting.

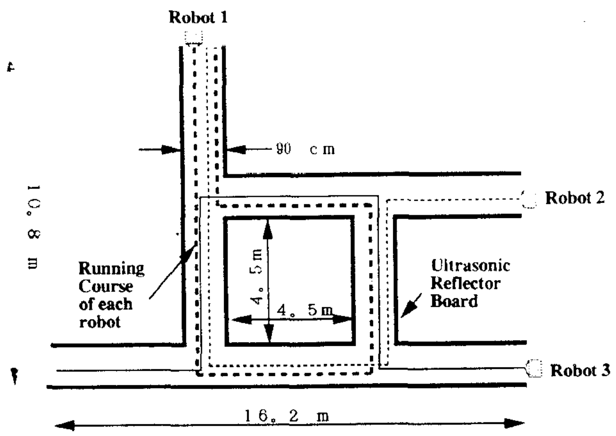

Fig. 3 Environment and running course of each robot

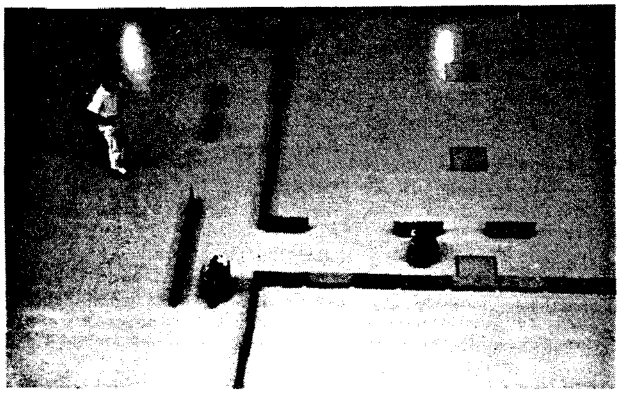

Photo 1 Show three robots running in the same environment

7. Conclusions

This paper considers a world with several self-contained autonomous robots operating in a closed space. Cooperation of the robots has been analysed. When several robots work close to each other, the fundamental cooperation among them is a collision avoidance of the spatial resource access. For this type of operation, the authors proposed the modest cooperation. This concept is useful for factory automation which many robots are working together cooperatively.

In this paper, it is assumed that robots in a system have almost the same functions and abilities, and the robots also move with almost the same algorithm. In the model of human society, this assumption is mostly effective. In another more generalized model which consists of various species of animals such as men, dogs, ants, etc.; problems of cooperation will be more complicated. Problems concerning a cooperation of robots with different features like such model is also an attractive subject to be examined in the future.

Acknowledgments

The Yamabico Project is running with strong support from its members who are mainly located at the Intelligent Robot Laboratory at University of Tsukuba. Dr. Jun' ichi Iijima of Meisei University is a key person on the Yamabico project. Dr. Shigeki Iida and Mr. Shooji Suzuki have contributed greatly to the route runner algorithm and the programming environment. Mr.Yasunori Abe of Shinryo-reinetsu Co.Ltd. helped to provide the experiment site. Dr. Alex Zelinsky of the University of Wollongong, Australia gave valuable suggestions in preparing and proof reading this paper. The authors thank the above mentioned people for their contributions.

References

[1] S. Yuta,'Autonomous Self-contained Robot''Yamabico' and its Controller Architecture', Third Australia National Conference on Robotics, Melbourne,1990.

[2] S. Premvuti and S. Yuta, 'Radio Communication Network on Autonomous Mobile Robots for Cooperative Motions', Proc. IECON'88, Singapore, 1988.

[3] S. Suzuki, M.K. Habib, J. Iijima, S. Yuta, 'How to describe the mobile robot's sensor based behavior?', Intelligent Autonomous Systems 2, Amsterdam,1989.