

Formation and Cooperation for SWARMed Intelligent Robots

Wei Cao¹ Yanqing Gao² Jason Robert Mace³
(West Virginia University¹ University of Arizona² Energy Corp. of America³)

Abstract

This article discusses the feature of intelligent robots, and specially emphasizes the difference between the intelligent robot and the traditional mobile robot. The behavior of a Swarmed group of intelligent robots is presented. A new Master-Slave management model has been proposed by the authors and the developed Client-Server communication protocol is reported.

Section I: Intelligent Robots and Mobile Robots

An intelligent robot is different to the traditional mobile robot. The traditional mobile robot uses a powerful microcontroller to control the movement, with the help of sensing devices it normally is able to make a tactic decision but could not do any sophisticate thinking since the limitation of the computation power provided by microcontroller. The left of Fig.1 shows the Stamp Bot which is equipped with Stamp microcontroller and light, touch and other sensors.

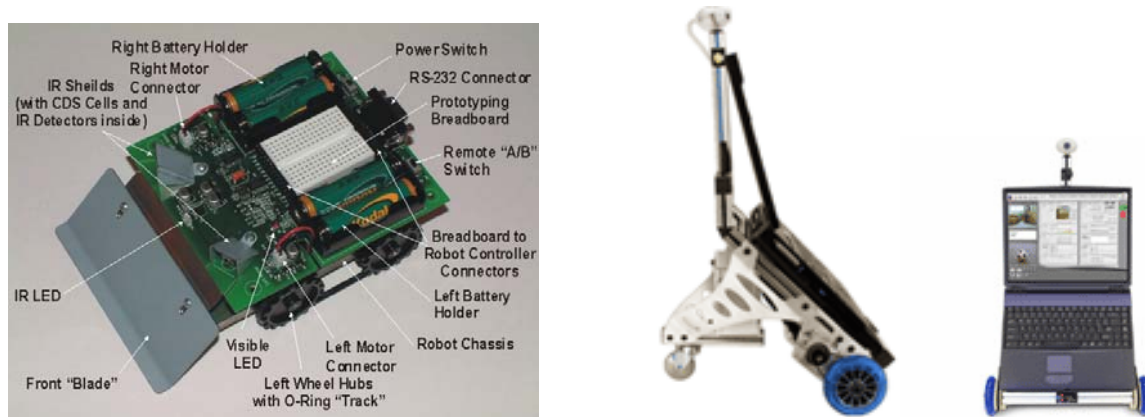
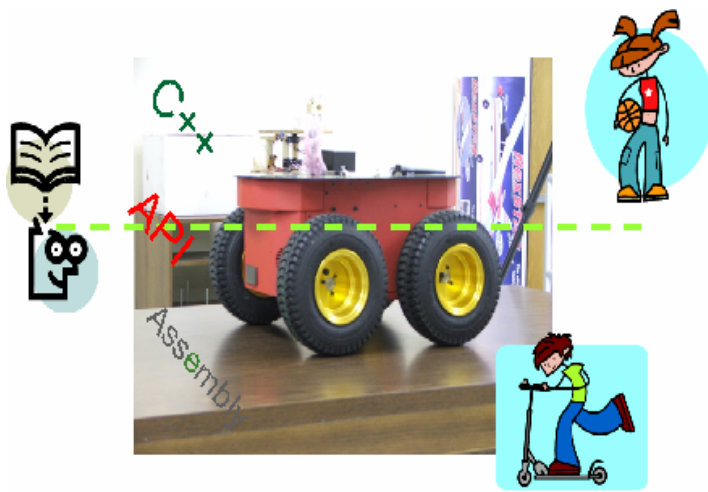


Fig.1 Traditional Mobile Robot (left) and Intelligent Robot (right)

However, in order to enhance the capability, the intelligent robot adds a separate layer of computation core, such as a laptop or a single motherboard, on the top of the movement control microcontroller. That means, by adding a "brain" the traditional mobile robot "body" is upgraded to an intelligent machine, which is capable to undertake many sophisticate environment sensing and decision making. The right of Fig.1 shows the ER1, which is a simplest model of intelligent robot. It adds a laptop on the top of the microcontroller. With full computation power of a laptop, more sophisticated and more accurate sensing information can be collected and processed. ER1 gains the capability to conduct intelligent activities, such as computer vision, voice hearing, speech processing, pattern recognition, networking, remote control, email, autonomous mobility, video transmission, physical gripping and etc



The typical structure of the intelligent robot is shown on Fig.2. It is seen that the bottom layer is with a powerful microprocessor, with traditional assembler for compiling and then the machine code is used to directly control the basic movements, such as translation and rotation with adjustable parameter (speed, acceleration, distance, angle and etc). A deliberate driving system with gyro and wheel encoder will work together to accomplish the precise movement.

Fig.2 Typical Structure of Intelligent Robot

For an intelligent robot, the top layer is a motherboard or laptop with either window or Linux operating system, which can fulfill all the intelligent works. Originally, those works are supposed to do on a powerful standalone laptop or desktop. The communication between the top layer and bottom layer is handled by API, which will interpret all the decisions to some assembler function, and then a series of basic movements will be issued.

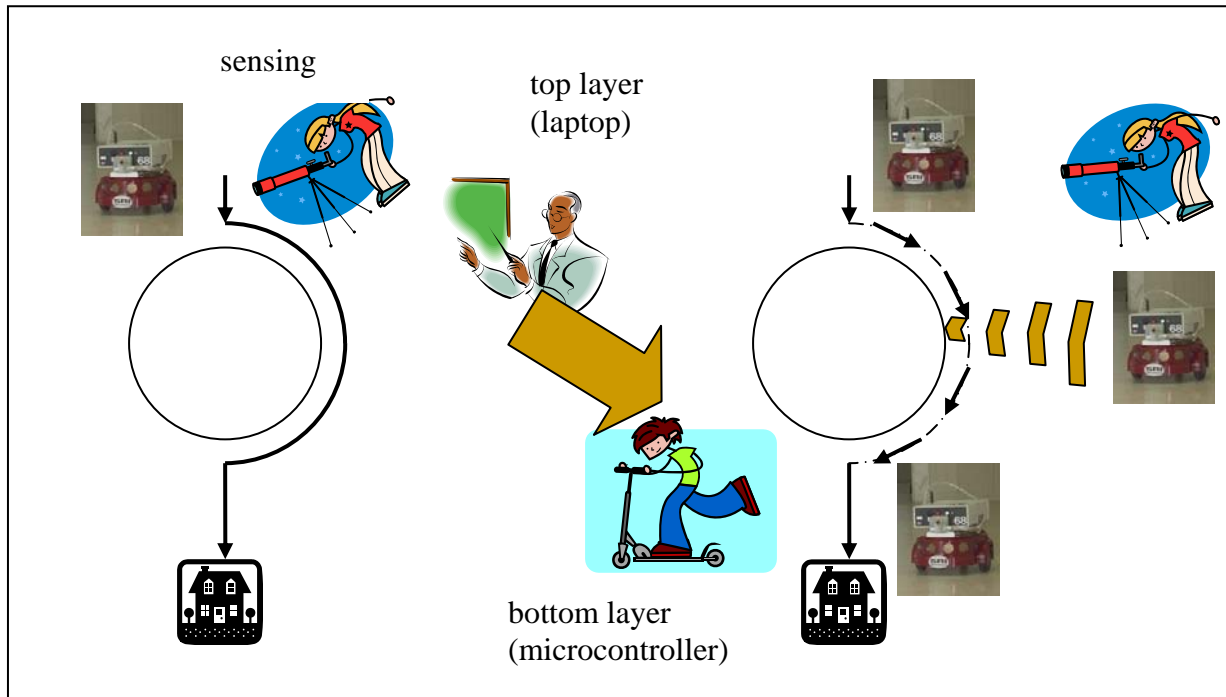


Fig.3 Intelligent behaviors

For example, Fig.3 shows an intelligent robot has to attack terrorists who hide in a house with hostages. When the sensing system finds an obstacle between, the top layer (brain) will calculate the optimum path, which is an arch around the object. Then the top layer will decompose the best path to a series of assembly routines (translational and rotational movements), and then the top layer will demand the bottom layer (body) to accomplish the movements. During the moving, a sophisticated sensing system (IR, Acoustics, Laser, Sonar, Camera, Rader and etc) will get the distance between the obstacle and other necessary parameters for robot to adjust the movement in real time.

It is known that the microcontroller is a very hardware dependent device without general operating system, such as Window and Linux does. The one of the most critical advantages to add core of Window or Linux system on the top of the microcontroller is that the developer can adapt any advanced intelligent commercial package into the intelligent robots system almost in no time. Just load commercial package into the top layer (with either Window or Linux platform) and design a proper interface (depending on the application) to communicate with the bottom layer. In that way, the development cycle of the intelligent robots has been reduced significantly to minimum. For example, there are many high-end software packages for video transmission and processing on the market. The only thing we have to do is to pick the one closely matching your application and write an API or the similar interface to integrate and synchronize with bottom layer, which control the body movement. Another critical example is, you do not have to write all the low-level communication protocols by yourself (like you do for any microcontroller if the communication to others is necessary). Since the built-in TCP/IP protocol for Window and Linux pretty well take care much of the main tasks. What you have to do is to write those special protocols fitting your own application.

Section II: Group Swarmed Intelligent Robots

During the past decade, many algorithms and experiments have been conducted for intelligent robots [1-4]. For example, the intelligent behavior shown on Fig. 3 is the result of a very simple SLAM (Simultaneously Localization and Mapping) algorithm. It is fair to say, that currently the path planning and obstacle avoidance has been an essential built-in portion of any commercialized intelligent robot. Many of those intelligent robots have been manufactured in a quite large amount and used in battle field, such as explosive detection and etc.



Fig.4 Swarmed Intelligent Robots

However, regarding to the formation and cooperation among a group of intelligent robots there are still many open questions existing now, which attract many attentions from the researchers. Among

them, “swarming” is an emerged popular subject (see Fig.4). Conceptually, “swarming” means that a huge number of intelligent robots with verity of different operation platforms work together as a swarming group; the organization and function of a swarming team is just like bees or ants. The individual intelligent robot can run in either autonomous mode or cooperative mode. Normally, there is one or more ground station(s) to coordinate and initiate the swarming team. The path planning and obstacle avoidance will become a part of formatted cooperative team work.

The communication between the ground station(s) and individual intelligent robots has been developed in a systematic manner in the past decade. However, there is no convinced and reliable physical communication means between individual robots available. And the fact of that there isn’t any methodology of information exchanging between individual intelligent robots is commonly recognized among researchers

Section III: Master-Slave Management Model

The authors of this article proposed a new Master-Slave approach to handle the cooperative management between intelligent robots. In the proposed setting (see Fig.5) one or a few of intelligent robots, which is with high-end on-board computation power and virtual link to ground station(s) will play the role of Master. The intelligent management algorithm will be resided on the Master as well on ground station(s). The ground station link will be activated if the complexity of the computation is beyond the power of Master intelligent robots, and the computation will be conducted on ground station(s).



Fig.5 Mater-Slave Intelligent Robots

That setting is very similar to the relationship between space shuttle and space ground station. All other intelligent robots, which are with limited computation power and function, will be slaves. Those Slave intelligent robots will receive the updated commands from Master on the time schedule. Based on the real surrounding environment, the intelligent robots will conduct individual or grouped tasks.

Master-Slave group is formed a task cluster (see Fig.6). Master intelligent robots are assumed to be retrievable since it might be a facility with a high cost. And the most of the tasks (bombing, surveying and etc) will be accomplished by slaved intelligent robots. Slave intelligent robots are with low cost and

can even be destroyed after the accomplishing the task. It is seen that the wireless communication based on the TCP/IP protocol is built-in with our Master-Slave Linux based cluster.

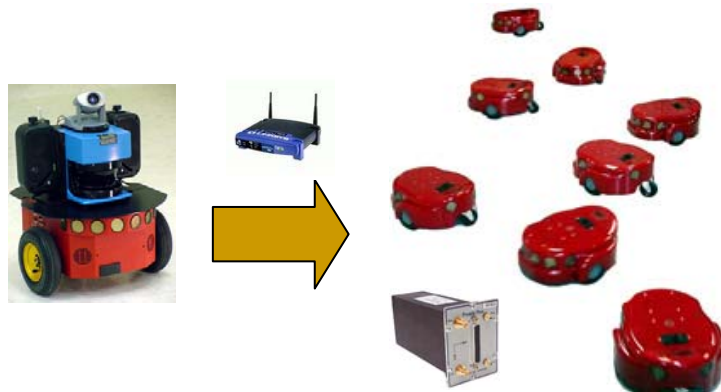


Fig.6 Master-Slave Task Cluster

Section IV: Client-Server Communication Protocol and Simulator

As we know the communications between members of the Swam team is very dynamic. For example, the Master can issue an assignment for several Slaves, and then it will leave all Slaves alone to accomplish the task without contacting after their dispatch. The communication between Slaves in the group could be kept minimum or one of the Slave will take the role of intermit commander. Or some tasks require a Master on the touch base to communicate with the ground station all the time or most of the time. Or several Masters have to cooperatively work together and etc.

Therefore the client-server protocol is with the most flexibility and reliability. In the past several years, our team has developed a library of basic client-server communication protocols, which is particularly useful for the communications between Mater-Slave, Master-Master and Slave-Slave. Beside that, we have developed a multi-robots simulator in the house. This simulator is capable to simulator up to 10 robots with variety of setting for Mater-Slave, Master-Master or Slave-Slave cases. All the protocols have been tested using this multi-robots simulator.

Combining with the adapted GroundStation-Master communication protocol from other party, a prototype version of the intelligent management algorithm for swarmed intelligent robots has been developed successfully, which has been simulated using Matlab and it is currently tested and evaluated using advanced swarmed multi-intelligent robots system.

- [1] G. DeSouza, A. Kak, "Vision for Mobile Robot Navigation: A Survey", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 24, No. 2, February 2002.
- [2] Kanade, T., Amidi, O., and Ke, Q., "Real-Time and 3D Vision for Autonomous Small and Micro Air Vehicles", *IEEE Conference on Decision and Control*, Paradise Island, Bahamas, December 2004.
- [3] A. Kurdila, M. Zabaranin, O. Prokopyev, A. Goel, V. Boginski, R. Causey, S. Uryasev, P. Pardalos, "Vision-Based Trajectory Planning for Autonomous Micro-Air Vehicles in Urban Environments", Research Report # 2003-001, December, 31 2003.
- [4] A. Richards, Y. Kuwata, J. How, "Experimental Demonstrations of Real-time MILP Control", *AIAA Guidance, Navigation, and Control Conference and Exhibit*, Austin, Texas, 11-14 August 2003.

Appendix: Flow Chart of Proposed Master-Slave Management Model for Swarm Team

