



Journal of Applied Sciences

ISSN 1812-5654

science
alert

ANSI*net*
an open access publisher
<http://ansinet.com>

A Novel Method for Controlling Multi-agent Robot

Shadi Montazer Rahmati and Farshad Farhadnia
IT Center, University of Tabriz, Tabriz, Iran

Abstract: In this study Multi-Agent systems and their control process are introduced. We concentrate on Khepera II robots as a Multi-Agent System and propose a new developed method based on Braitenberg procedure. The most important aspect of this method is utilizing Braitenberg theory for two robots based on impedance control. Our case study is consisted of two Khepera II robots which move on a plane including various obstacles. One of these robots (follower) must track another one (leader) on this plane by sensory connectivity based on Impedance Control (IC) strategy. We control these robots via two heterogeneous networks; the leader robot is connected to a wireless network. The simulation process is performed by Webots® software and the results show the effectiveness and high performance of this new algorithm.

Key words: Control, multi-agent robot, heterogeneous network, khepera robot, webots software

INTRODUCTION

Now-a-days, use of multi-agent systems has been attended more se of utilization of these systems in various scientific fields. For instance, it can be used in military and security services, prediction of economical status, social problems and etc. (Axelrod, 1984; Braitenberg and Zadro, 1999).

However, the controlling process which is carried out by robot, depends on the ability of robot components, movement environment, light, color, type of sensor and data transaction, clarity of area and communication of robot and controller (Isard, 2003; Steels, 2004).

During the last few years there has been a growing interest on coordination and control of multiple mobile robots. In the leader-follower approach a robot is designed as the leader while the other is the follower (Braitenberg *et al.*, 2006; Sisto and Gu, 2006). The leader moves along predefined trajectories while the follower has to follow the leader while having a desired relative posture (distance and angle) with it. It is important to notify that both Leader and Follower have to be independent from various obstacles. By leader-follower approach we only need to specify leader's motion and the desired relative posture between leader and follower.

Among all the approaches to control mobile robots reported in this study, the leader-following method has been adopted by Burgsteiner (2005), Isard (2003) and On *et al.* (2008). In this method, a robot takes another robot as a reference point to determine its motion. The referenced robot is called leader and the following robot is called follower.

Braitenberg *et al.* (2006) proposed a series of thought experiments involving the testing and construction of

simple autonomous robots of increasing complexity. His book added a fresh perspective to the field of robotics because his vehicles were based on ideas from Artificial Neural Networks (ANNs). We have used Braitenberg's Vehicles as a framework to control two Khepera II robots as a Leader-Follower system (Braitenberg *et al.*, 2006).

This study deals with an investigation for controlling two robots by both wireless and cable systems. The aim of this research was to design a controller to make communication between two heterogeneous networks according to specific target to control follower robot by the leader robot. Various factors like walls, color, solid objects have effect on the movement plan of robot. In this study we used Khepera II robot because of its capability. It has 8 sensors, ability of movement in different direction, proper dynamic and possibility of connecting to network cable and wireless at the same time. In order to reduce the error, two computers have been installed as a local server to control the robots and a computer as a main controller. Webots® software has been used to control the robots. This software is able to control many robots at the same time and has got proper interface.

MATERIALS AND METHODS

System architecture: Our experiments were performed with a standard autonomous miniature robot, the Swiss mobile robot platform Khepera II (Braitenberg and Zadro, 1999). It is equipped with eight infrared proximity sensors. The mobile robot has a circular shape, a diameter of 6 cm and a height of 5 cm. It possesses two motors and an on-board power supply. The motors can be controlled independently by a PID controller. The eight infrared

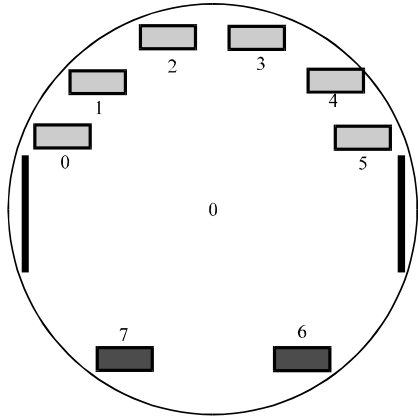


Fig. 1: Khepera II robot

sensors are distributed around the robot in a circular pattern. They emit infrared light, receive the reflected light and measure distances in a short range: of 2-5 cm. It is possible to control the robot in two ways. The controlling algorithm can be executed on a workstation with data and commands communicated through the serial line. Alternatively, the controlling algorithm is cross-compiled on the workstation and down-loaded to the robot which runs the complete system autonomously. At present we use both versions of the system (Wang *et al.*, 2000). Figure 1 shows the robot, its sensor placement and the sensor range in its environment.

Control algorithms: A series of thought experiments is described in which vehicles with simple internal structure behave in unexpectedly complex ways (Braitenberg, 1984; Chen *et al.*, 2003). He describes simple control mechanisms that generate behaviors that we might call aggression, love, foresight and even optimism, if we did not already know the principles behind the vehicles' operation. Braitenberg gives this as evidence for the "law of uphill analysis and downhill invention," meaning that it is much more difficult to try to guess internal structure just from the observation of behavior than it is to create the structure that gives the behavior.

Braitenberg vehicles: Braitenberg vehicles are simple automatons conceived in thought experiments proposed by the German cyberneticist Valentino Braitenberg to illustrate the abilities of reactive agents, thus representing the simplest form of behavior based on artificial intelligence or embodied cognition, i.e., intelligent behavior that emerges purely from motor sensor interaction between the agent and its environment without any need for an internal memory, representation of the environment or inference (Burgsteiner, 2005; Geman *et al.*, 2002).

The weight distribution of the robot sensors ray is equal with the Gaussian distribution:

$$\omega_i = \frac{\exp\left(-\frac{t_i^2}{a.g}\right)}{\sum_{i=1}^n \omega_i} \quad (1)$$

where ω_i is the weight of the i^{th} ray, t_i is the angle between the i^{th} ray and the sensor axis, α is the aperture angle of the sensor, g is the Gaussian width and n is the number of rays.

The schematic of Khepera II robot and the relation among the sensors and the robot's wheels is as follows (Fig. 2).

Mechanism: A Braitenberg vehicle is an automaton that can autonomously move around. Depending on how sensors and wheels are connected, the vehicle exhibits different goal-oriented behaviors. This mechanism is brought in Chen *et al.* (2003).

Behavior: In a complex environment with several sources of light and shadow, Braitenberg vehicles will exhibit a complex and dynamic behavior. The behavior details are brought in Braitenberg (1984).

Preparation: Present method is described as a extended approach of typical Braitenberg algorithm. The Braitenberg algorithm is designed for only one robot. This algorithm is between sensors and robot which lead to optimized trajectory while in our case there are two robots and in order to apply Braitenberg method we define a relation among sixth and seventh sensors of Leader and second and third sensors of Follower. This procedure is implemented in Webots® software based on C++ language tool and the relation among various sensors is created by Proximity property. The very important condition in this strategy is the adaptation between environment light and obstacles color. The flowchart of developed algorithm is illustrated in Fig. 3.

Artificial Neural Network (ANN): Now, a connection must be created among sensors. To reach this goal, the relation between sensors 7, 8 (Leader) and sensors 2, 3 (follower) must be connected using proximity; this process is demonstrated in Fig. 4.

Regarding our performance in utilizing the braitenberg controller for routing (path planning) and following tasks of the leader and follower robots and since the data transferring should be done in the form of two heterogeneous network and considering this

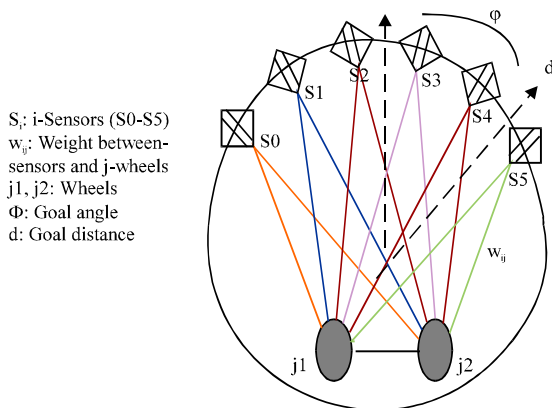


Fig. 2: The schematic of Khepera II Robot

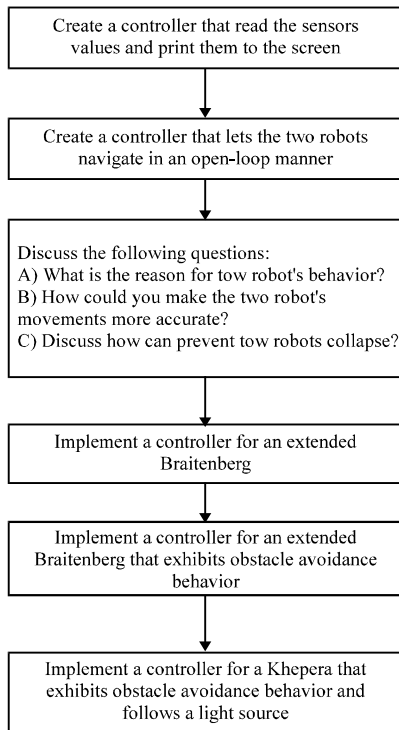


Fig. 3: The flowchart of developed algorithm

scenario as a multi-agent system we should apply proper learning algorithms to improve the path planning for the follower robot. Figure 5 shows the new configuration of sensors and wheels of the leader and follower.

For this reason and since the essence of braitenberg controller is based on artificial neural networks we use Back Propagation (BP) to form a learning process for the follower robot as the first step. So we sketch the problem to modify the path of the follower robot by optimizing the path and approaching to the desired points.

To utilize the artificial neural network, first we study its formation, the required functions and their capabilities and we simulate it then study the resulted errors. So, by

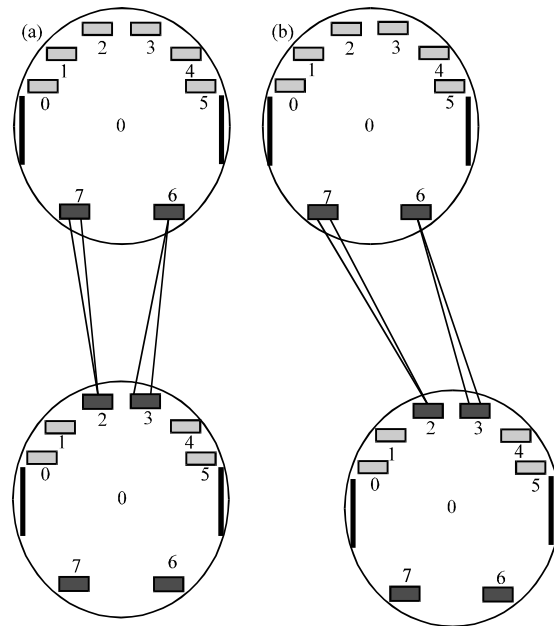


Fig. 4: The connection between Leader and follower; (a) Leader in front of the follower, precisely and (b) Leader in front of the follower, imprecisely

giving a summary of the method and introducing the required tools we consider the learning process and apply it to this problem.

There are several learning methods for artificial neural networks and for different functions such as functions with real values, discrete and vectored functions. In this research we will utilize the learning of discrete functions.

The learning process of ANN is not affected by the learning data error and they easily can be applied to the learning process of robots.

Here we use the traveled path by the follower robot to consider the error and desired points as the learning data. Since we need a learning process (stage) for the NN we use one of them which are built based on a calculation so-called perceptron unit. Given the real values of inputs, a perceptron calculates a linear composition of these inputs. If the result is more than a desired value, then its output is set to 1, otherwise it is -1. Figure 6 shows the conventional structure of a perceptron neural network with a hidden layer.

In our proposed model for the system with two robots which move in a planar (2-Dimension) surface with real values of (x_i, y_i) , i.e., the coordinates of the follower robot in each sample time and in the initial traveled path we can determine the desired coordinates of (x^i, y^j) .

Then using the gradient formula, the distance from each $A_i = (x_i, y_i)$ to the desired $A^i = (x^i, y^j)$ is calculated. If the result is greater than the threshold value, the output becomes 1, otherwise it becomes -1.

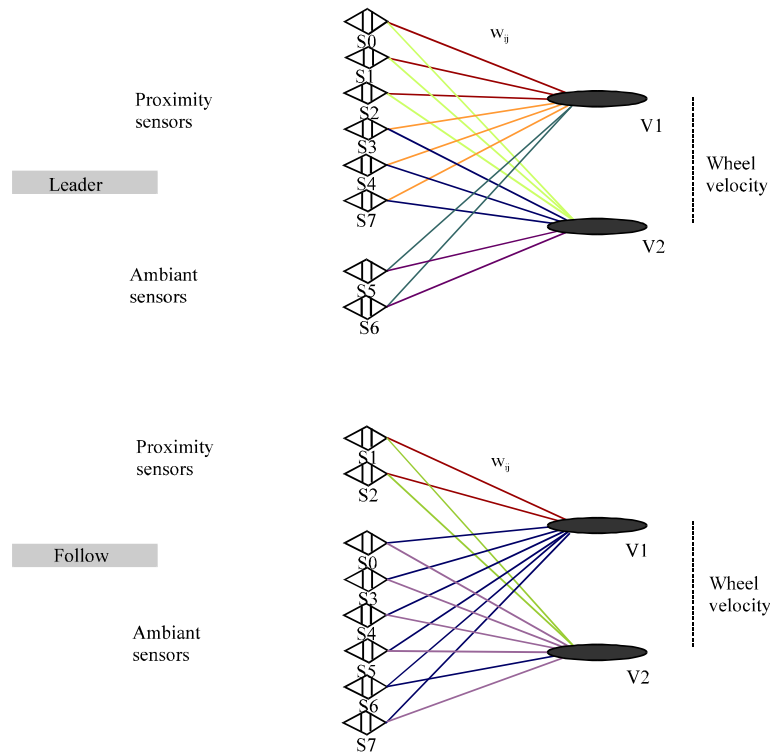


Fig. 5: New configuration between sensors and wheels

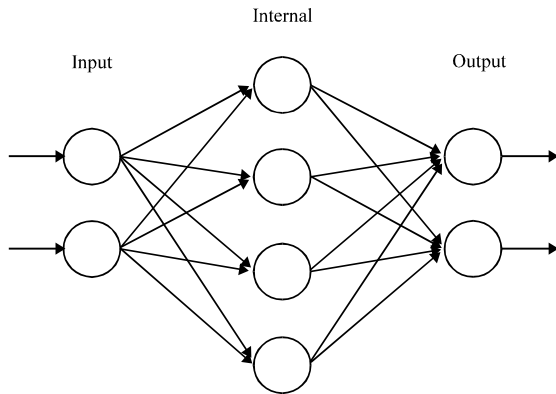


Fig. 6: Neural Network structure

Equation 2 represents the input vector for the Neural Network (NN). Equations 2 to 7, is demonstrated the learning process of the neural network:

$$x = (x_1, x_2, \dots, x_n) \quad (2)$$

Here, we consider the network with 2 inputs (x_1, x_2) :

$$\begin{aligned} x &= (x_1, x_2) \rightarrow x = (x_i, x_j) \\ \hat{x} &= (\hat{x}_1, \hat{x}_2) \rightarrow \hat{x} = (\hat{x}_i, \hat{y}_j) \end{aligned} \quad (3)$$

The weighting vector is as follows (Eq. 4, 5):

$$w_i = w_i + \Delta w_i \quad (4)$$

$$\begin{aligned} \Delta w_i &= \eta(t - o) \rightarrow \Delta w_i = \\ &\eta(t - o) \sqrt{(\hat{x}_i - x_i)^2 + (\hat{y}_j - y_j)^2} \end{aligned} \quad (5)$$

where, η , o and t are the learning factor, real output vector and output target, respectively.

The Eq. 6 and 7 are the gradient vector of error (E) and the Back Propagation method for learning process, respectively:

$$\nabla E(w) = \left[\frac{E'}{w_0}, \frac{E'}{w_1}, \dots, \frac{E'}{w_n} \right] \quad (6)$$

$$E(\bar{w}) = \frac{1}{2} \sum_{kd \in D} \sum_{ke \in \text{outputs}} (t_{kd} - o_{kd})^2 \quad (7)$$

SIMULATION RESULTS

Here, the simulation results for implementing extended controller are proposed. We use the WEBOTS[®] software based on C++ language for implementing the proposed method in this paper. This controller is exerted to Khepera II robot using the connections created among sensors of the Leader and Follower robots. Figure 7 shows the configuration of two robots before of motion.

The Fig. 8 illustrates the state of Leader and Follower before reaching the obstacle (until, there is not any path variation).

Figure 9 demonstrates the configuration of robots in the motion state while moving (Follower is varying its trajectory).

Figure 10 shows the desirable trajectory (line -) and real path (dash ---) of robots which consists of three

parts that compares three methods: our proposed method, Fuzzy and analytical, respectively.

As the Fig. 10 illustrates, the maximum error is happened in the starting point (the learning process is not complete) and while the learning stage is complemented,



Fig. 7: The configuration of robots before of motion

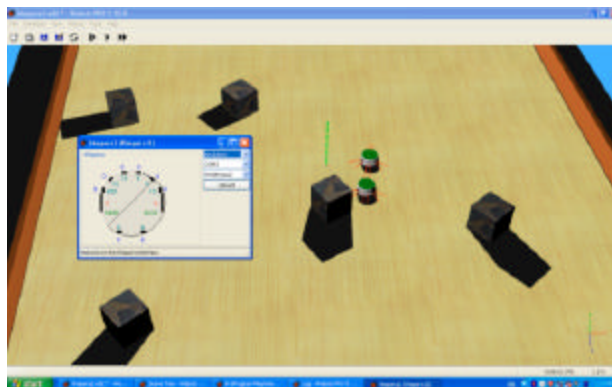


Fig. 8: The state of robots before of obstacle



Fig. 9: The configuration of robots in the stage of motion of next to the barrier

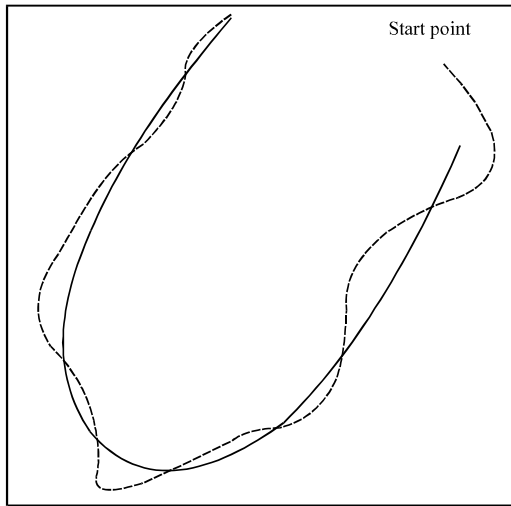


Fig. 10: The desirable trajectory and real path

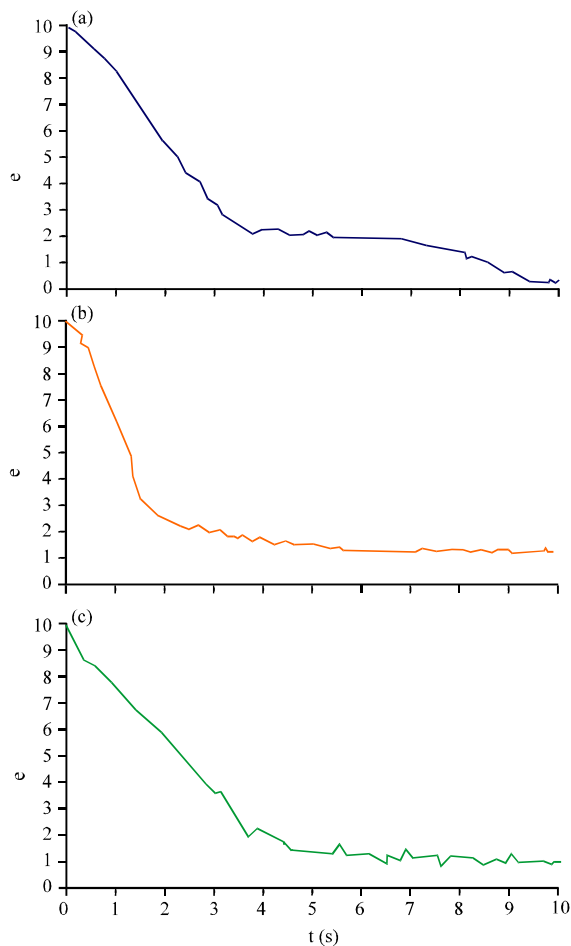


Fig. 11: The error function (convergence diagram); (a) Analytical method, (b) Fuzzy method and (c) Our proposed method

the value of error is decreased at the end. The convergence diagram (error function) is illustrated in Fig. 11a-c.

Regarding to this point that the follower movement is based on the tuning of proximity of the sensors, therefore the maximum of value error in entire of process (especially at first) is 10 centimeters. The sampling time is tuned on 100 msec.

Figure 11 illustrates that the analytical method finds the best trajectory compared with other approaches. Also the Fuzzy method has the good convergence at first but the final value of error function that is not satisfactory. Therefore, our proposed method in this study is an average manner of others.

CONCLUSION

In this study, we have developed the typical braitenberg algorithm to more robots (two robots in this study). Our case study is included two Khepera II robots which move on a plane including various obstacles and a robot (follower) must track another one (leader) on this plane by sensory connectivity based on Impedance Control (IC) strategy. The Braitenberg method is created for only one robot and there is an algorithm between sensors and robot which leads to appropriate path while in our case in this paper there is two robots and in order to exert Braitenberg algorithm we define a relation among sixth and seventh sensors of Leader and second and third sensors of Follower. This process is programmed in WEBOTS® software based on C++ language and the relation among different sensors is designed by Proximity property. The very noticeable condition in this application is the appropriate adjustment between environment light and obstacles color. The simulation results show the effectiveness of proposed procedure and we try to control these two Khepera II robots via two heterogeneous networks as a future study.

REFERENCES

Axelrod, R., 1984. *The Evolution of Cooperation*. Basic Books, New York.

Braitenberg, C. and M. Zadro, 1999. Iterative 3D gravity inversion with integration of seismologic data. *Bollettino Geofisica Teorica Applicata*, 40: 469-476.

Braitenberg, C., S. Wienecke and Y. Wang, 2006. Basement structures from satellite-derived gravity field: South China searing. *J. Geophys. Res.*, 111: 1-15.

Braitenberg, V., 1984. *Vehicles: Experiments on Synthetic Psychology*. MIT Press, Cambridge, Massachusetts.

- Burgsteiner, H., 2005. Training networks of biological realistic spiking neurons for real-time robot control. Proceedings of the 9th International Conference on Engineering Applications of Neural Networks (EANN'05), New Orleans, USA., pp: 4620-4626.
- Chen, X., A. Serrani and H. Ozboy, 2003. Control of leader-follower formations of terrestrial UAVs. *Decision Control*, 1: 498-503.
- Geman, S., D. Potter and Z. Chi, 2002. Composition systems. *Q. Applied Math.*, 60: 707-736.
- Isard, M., 2003. PAMPAS: Real-valued graphical models for computer vision. *Comput. Vision Pattern Recogn.*, 1: 613-620.
- On, C.K., J. Teo and A. Saudi, 2008. Evolving neural-based cognition of RF signals in autonomous Khepera II robots. Proceedings of the International Symposium on Information Technology, Aug. 26-28, Kuala Lumpur, pp: 1-8.
- Sisto, M. and D. Gu, 2006. A fuzzy leader-follower approach to formation control of multiple mobile robots. Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, Oct. 9-15, Beijing, China, pp: 2515-2520.
- Steels, L., 2004. Emergent functionality in robotic agents through on-line evolution. Proceedings of the 4th International Workshop on the Synthesis and Simulation of Living Systems Artificial Life IV, July 2004, MIT Press, Cambridge, MA, USA., pp: 8-16.
- Wang, L.F., K.C. Tan and V. Prahlaad, 2000. Developing Khepera robot applications in a Webots environment. Proceedings of International Symposium on Micromechatronics and Human Science, Oct. 2000, Nagoya, pp: 71-76.