Relationship between Quality of Control and Quality of Service in mobile robot navigation

José-Luis Poza-Luján, Juan-Luis Posadas-Yagüe, and José-Enrique Simó-Ten

University Institute of Control Systems and Industrial Computing (ai2). Universitat Politècnica de València (UPV). Camino de vera, s/n. 46022 Valencia (Spain).
{jopolu, jposadas, jsimo}@ai2.upv.es

Abstract This article presents the experimental work developed to test the viability and to measure the efficiency of an intelligent control distributed architecture. To do this, a simulated navigation scenario of Braitenberg vehicles has been developed. To test the efficiency, the architecture uses the performance as QoS parameter. The measuring of the quality of the navigation is done through the ITAE QoC parameter. Tested scenarios are: an environment without QoS and QoC managing, an environment with a relevant message filtering and an environment with a predictive filtering by the type of control. The results obtained show that some of the processing performed in the control nodes can be moved to the middleware to optimize the robot navigation.

1. Introduction

In mobile robot navigation architectures, different components work at different control nodes that are connected through the communications channels. To measure the efficiency of the communications, and the quality of component's services, system uses the concept of Quality of Service (QoS) [1] through the QoS parameters [2]. The communications management oriented to optimize the QoS parameters is known as QoS policies [3]. Among standards to manage distributed communications systems, the DDS standard [4] implements a large type of QoS policies. DDS is based on publish-subscribe paradigm, extended with some elements that connect the application synchronously (readers and writers) and asynchronous (listeners). A good explanation of the operation can be found at [5]. Therefore, DDS is well suited for implementing distributed intelligent control architectures [6].

To measure the control efficiency, currently is used the concept of Quality of Control (QoC) [7]. The QoC measures the quality of the control action through equations, generally using the difference between the input signal and the reference signal. Sometimes the QoC parameters are used as feedback of control action; thus, the QoC measures the control efficiency and it makes easier the control processing.

The control efficiency does not depend exclusively on the algorithms used; the communications efficiency also affects the control action. [8]. To prove the relationship between QoS and QoC, an architecture called FSACtrl [9] and [10] has been developed. FSACtrl allows measuring QoC and QoS parameters in control nodes. Architecture is based on DDS standard, and it uses the DDS QoS policies to manage the communications.

Paper describes tests performed in a simulated mobile robot environment. It shows results obtained by using QoS and QoC to measure the efficiency of control node depending on the communications configuration.

The paper is organized as follows: the following section describes the environment used to perform tests of the architecture: simulation environment and simulated robots. Then, third and four sections describe the QoC and QoS parameters that have been considered in the described environment. The fifth section describes tests performed and results. Finally, the paper ends with conclusions of experiments done and the future work to be developed.

2. Experimental environment

To test the architecture, the control of first five Braitenberg vehicles [11] has been simulated. The first three vehicles are characterized by the lack of advanced control functions; so that, these vehicles are suitable for evaluating the performance of the communications because messages are processed principally in the middleware.

The interest of Braitenberg vehicles is in the simplicity of control, based on the simple functions that connect sensors and actuators. In addition, the possibility to have different types of sensors that react to different sources provides a lot of messages that are used to test the effect of communications configuration in the control efficiency.

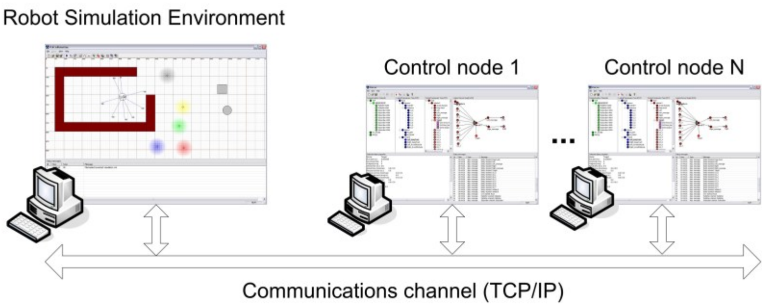

Fig. 1. Experimental environment used to test the FSACtrl architecture.

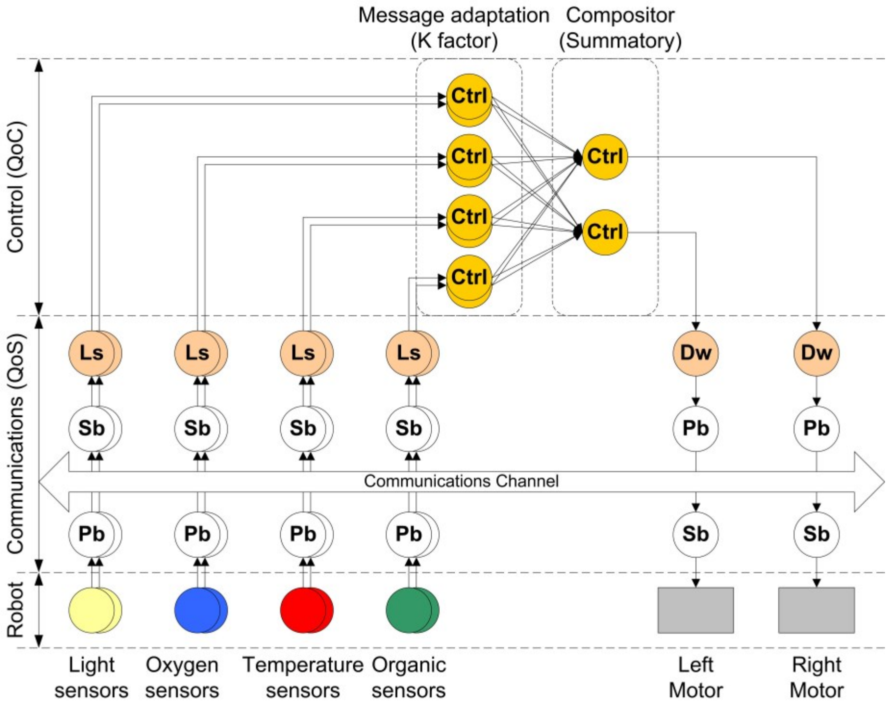

Fig. 2. FSACtrl architecture implementation of the Braitenberg 3.c vehicle.

To test the architecture, a simulation platform has been developed. The platform has a mobile robot simulator and an application to design and implement the control algorithms. Figure 1 shows the topology of the distributed system used to test the architecture.

The simulator allows user to create a 2D environment and insert any number of robots. For each robot, the simulator has twelve different types of sensors. All robots are circular and have two motors. This configuration allows robots to move in any direction in the simulated environment.

The robot simulation environment is composed of a space with different signal sources and rectangular and circular obstacles. The simulator sends via TCP clients the data from the sensors of each robot, and it receives, via a TCP server, speeds assigned to each robot motor.

Control nodes are composed of an FSACtrl elements editor that launches the control processes. The editor allows insert, modify and configure QoS policies and QoS parameters to each FSACtrl architecture element. The system implements the control node over personal computers on a TCP/IP based network. The accuracy of the measurements in the control nodes is nanoseconds; the computation time of the control nodes has been simulated in order to obtain comparable results.

3. Quality of Control in robot navigation

Braitenberg vehicle that uses simple control algorithms and works with a large amount of data is the vehicle 3.c. FSACtrl architecture elements are shown in figure 2. The vehicle calculates the direction that should be taken based on information obtained from the four types of sensors available, using equation 1.

$$Output_{compositor} = \sum_{i=1}^{N} K_i \cdot Input_i \tag{1}$$

The output of each of the composers is calculated from the contribution of each input of the N sensors of the vehicle, weighted by a specific K factor for each sensor. The quality of control in the 3.c Braitenberg vehicle is measured by means of the angle that the vehicle deviates from the planned angle in the theoretical analysis of the vehicle mission (figure 3). The equations to obtain the quality of control parameter can be very different, because of the quality parameter is directly associated to the characteristics of the robot on which it is applied [12]. In the case of vehicle 3.c the quality of control is directly calculated with the parameter ITAE (equation 2)

$$ITAE = \int_{t_0}^{t_{END}} t \cdot \left| \varphi_y(t) - \varphi_r(t) \right| dt \tag{2}$$

In the equation 2, $\varphi_y(t)$ is the value of the desired angle for a time t, while $\varphi_r(t)$ is the real angle obtained in the same instant of time. ITAE parameter considers the navigation error with the same weight during all the navigation time, so that it is very suitable to make global comparisons. The smaller ITAE value, the better quality of navigation of the vehicle is. This is because of the angle obtained from the course is closer than expected angle.

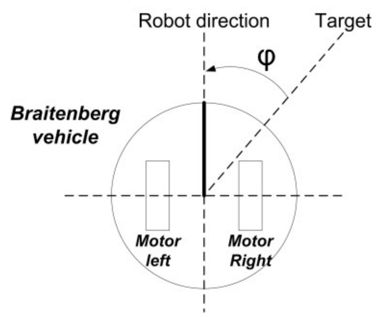

Fig. 3. Path error in the Braitenberg 3.c vehicle used to calculate the ITAE parameter.

4. Quality of Service in robot navigation

Middleware manages the QoS. In the case of FSACtrl architecture, QoS is managed by the QoS policies of the DDS standard. In the tests performed the QoS parameters that have been measured are the control component load and the rate of useful messages. The control component load (ρ) is calculated as the rate between the service demand and the service rate of the component. Due to the architecture elements are made by messages queues [13], global load is obtained through the pondered rate of each element load (equation 3). The K factor is used to balance the most important control components.

$$\rho_{global} = \sum_{i=1}^{N} (K_i \cdot \rho_i) \Big/ N \tag{3}$$

To calculate the load ρ of each component is used the equation 4, where λ is the demand for the services requested from the vehicle control and μ is the rate of service provided by the control component. Both of these parameters are expressed in messages per second so that the load is a dimensionless parameter. Closer load to zero better is the control component load.

$$\rho = \lambda / \mu \tag{4}$$

The useful messages rate (*UM*) is obtained by means the equation 5. The concept of utility of a message can be quite large. In the experimental environment a useful message is considered when the message produces a change in vehicle navigation. The variation of navigation is produced when the control action calculated for a measurement is different from the control action calculated for the previous measurement. Closer to one are, better the parameter is. The control action in Braitenberg vehicles is performed on the speed of the motors.

$$UM = N_{output\ (i) \neq output\ (i-1)} \Big/ N_{total} \tag{5}$$

From the two previous equations, the performance (η) of the control can be obtained (equation 6). Performance is defined as the satisfactory results obtained in relation to the cost in resources used. The control performance is obtained through the parameters from the equations 4 and 5. Through the performance equation, can be verified the effectiveness of the control messages related to the resources consumed from the control service.

$$\eta = UM \cdot (1 - \rho_{global}) \tag{6}$$

5. Experimental tests and results

Three scenarios on the architecture with the Braitenberg vehicle 3.c have been tested:

1. Control action without filtered messages optimization and without messages selection optimization (not QoS and not QoC management).
2. Control action with filtered messages optimization and without messages selection optimization (QoS managed but not QoC management).
3. Control action with filtered messages optimization and with messages selection optimization (QoS and QoC managed).

Message filtering consists of transmit through the middleware only those messages whose content is different, compared with the preceding message. The message filtering is one of the characteristics specified in DDS standard recommendations for a middleware. The control optimization is performed by inserting control components that predict the change in control action. The prediction is made comparing the messages from different sensors involved in the calculation of control action. The environment is a system without obstacle with the four types of sources associated with the four types of sensors of the vehicle 3.c of Braitenberg (figure 2). The vehicle is configured to be attracted by light and organic matter sources, and to be rejected by heat and oxygen sources. The vehicle follows a path that depends on the location of the sources in the environment (figure 4).

Tests have been performed starting the vehicle in the same position and the sources placed in the same location and changing the middleware according to each scenario described.

Table 1 shows experimental values for each of the scenarios described at the beginning of the paragraph. Columns show the average values of the control load, the usefulness of messages rate, the performance of the control element and the value of ITAE. Each row contains the data for each of the scenarios described above.

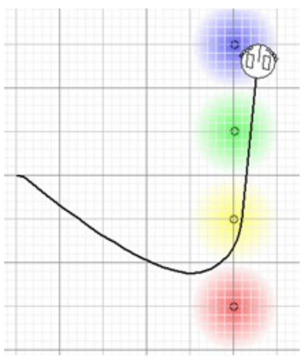

Fig. 4. Example of 3.c Braitenberg vehicle navigation in a multi-source environment.

Table 1. Experimental results based on different scenarios (average values).

Scenarios	ρ	UM	η	ITAE
1. Without QoS and QoC management	0,184	0,212	0,173	0,252
2. With QoS management and without QoC management	0,121	0,323	0,284	0,261
3. With QoS and QoC management	0,119	0,683	0,602	0,284

Due to the response time of control service is the same in all scenarios tested the variation of the control load depends on the message arrival frequency. Because of the scenarios 2 and 3 include a message filtering phase the control load decreases significantly respect the scenario 1.

UM rate changes progressively among the three different scenarios. In the scenario 2, UM value rises respect the scenario 1 because the middleware has filtered some messages that do not generate a control action. However, the most significant improvement of useful message index is produced in the scenario 3. In the scenario 3, the control receives only messages that haven't been filtered in the middleware and in the control prediction. For this reason the message utility rate increases considerably compared with the previous two scenarios.

Figure 5 shows the comparison between the service performance index (η) and the control index (ITAE). The service performance describes the common contributions of the two parameters analysed and it is a good measure of the quality of service that the control component provides. The figure shows how performance is directly related to the optimizations used in each scenario. ITAE parameter is used to check the efficiency of the control service optimizations of the vehicle navigation. In this case, ITAE parameter increases very slightly in relation with the optimized scenario, so that improvements implemented on every scenario do not affect the quality of the robot navigation mission.

6. Conclusions and future work

Results of the experimental work carried out are satisfactory. Results show that the FSACtrl architecture is viable as a middleware with support to simple control actions. It is also proves as manager of the communications layer allows to optimize the control layer that affects overall system optimization.

As future work, several studies related with the relation between QoS and QoC can be performed. One of the most interesting questions, to develop, is the dynamic adjustment, through QoS policies, of the robot navigation. The concept of the dynamic variation can be extended to the QoC with the QoC policies. The objective is determine the convenience to adjust the communications and control characteristics, as the sampling frequency, according to certain environmental and design constraints such as energy consumption or the time to complete the mission of the vehicle.

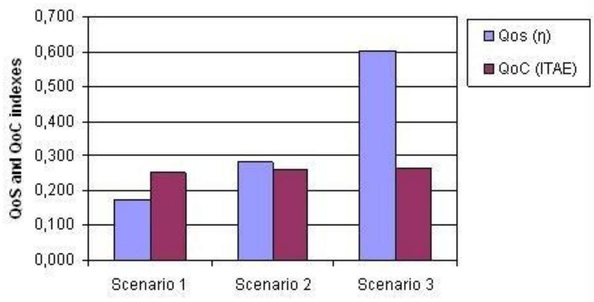

Fig. 5. Comparison chart between the η values (QoS) and the ITAE values (QoC).

Acknowledgments The work described in this article is a part of the coordinated project SIDIRELI: (Distributed Systems with Limited Resources) and COBAMI (Mission-Based Control) Education and Science Department, Spanish Government and European FEDER found. MICINN CICYT: SIDIRELI: DPI2008-06737-C02-01/02, COBAMI: DPI2011-28507-C02-02.

References

1. Vogel, A., Kerherve, B., von Bochmann, G. and Gecsei, J. *Distributed Multimedia and QoS: A Survey*. *IEEE Multimedia*.Vol.2, No. 2, pp. 10-19. 1995.
2. Crawley, E., Nair, R., Rajagopalan, B. RFC 2386: A Framework for QoS-based Routing in the Internet. *IETF Internet Draft*. pp. 1-37. 1998
3. Bradner, S. RFC 2026: The Internet Standards Process. *IETF Internet Draft*. section. 10. 1996.
4. Object Management Group (OMG): Data Distribution Service for Real-Time Systems, v1.1. Document formal / 2005-12-04. (2005)
5. Poza, J.L., Posadas, J.L. and Simó, J.E. QoS-based middleware architecture for distributed control systems. International Symposium on Distributed Computing and Artificial Intelligence. DCAI. Salamanca (Spain) 2008
6. Poza, J.L, Posadas, J.L. Simó, J.E. A Survey on Quality of Service Support on Middleware-Based Distributed Messaging Systems Used in Multi Agent Systems. DCAI. 9th International Conference on Practical Applications of Agents and Multi-Agent Systems. Salamanca (Spain) 2011.
7. R.C. Dorf and R.H. Bishop, Modern Control Systems, 11th Edition, Prentice Hall (2008)
8. Soucek S., Sauter T.,"Quality of Service Concerns in IPBased Control Systems", IEEE Transactions on Industrial Electronics, Vol. 51, No. 6, December 2004.
10. Poza, J.L, Posadas, J.L. Simó, J.E. Multi-Agent Architecture with Support to Quality of Service and Quality of Control. 11 th International Conference on Intelligent Data Engineering and Automated Learning. Paisley, UK. 2010.
11. Braitenberg, V., 1984. Vehicles: Experiments on Synthetic Psychology. MIT Press, Cambridge, Massachusetts.
12. Gabel O., Litz L., "QoS-adaptive Control in NCS with Variable Delays and Packet Losses – A Heuristic Approach", 43rd IEEE Conference on Decision and Control, 2004.
13. Poza, J.L., Posadas, J.L., Simó, J.E.: From the Queue to the Quality of Service Policy: A Middleware Implementation. In: S. Omatu et al. (Eds.): IWANN 2009, Part II, LNCS 5518, pp. 432–437 (2009)