15th Triennial World Congress, Barcelona, Spain

A NEW MOBILE ROBOT CONTROL ARCHITECTURE VIA CONTROL OUTPUT FUSION

Eduardo Oliveira Freire[1,2]
Teodiano Freire Bastos Filho[2]
Mário Sarcinelli Filho[2]
Ricardo Carelli[3]
Oscar Nasisi[3]

1 Grupo de Pesquisa em Automação, Instituto de Tecnol. e Pesquisa, Univ. Tiradentes
Av. Murilo Dantas, 300, Farolândia, 49032-490, Aracaju/SE BRAZIL

2 Departamento de Engenharia Elétrica, Universidade Federal do Espírito Santo
Av. Fernando Ferrari, s/n 29060-900 Vitória/ES BRAZIL

3 Instituto de Automática, Universidad Nacional de San Juan
Av. San Martin Oeste, 1112, 5400, San Juan, Argentina

Abstract: A new architecture for controlling the navigation of a mobile robot based on the fusion of the output of several controllers is proposed. A decentralised information filter accomplishes the fusion of the outputs of the controllers. The output of each controller is connected to a local filter with a covariance associated to it. The lower this covariance is, the bigger is the influence of the corresponding controller on the fused output. A fuzzy logic-based approach is proposed to determine such covariances. The control system is implemented in a commercial robot and its performance is shown through a practical navigation experiment. *Copyright © 2002 IFAC*

Keywords: Mobile robots; Autonomous vehicles; Robot control; Data fusion; Kalman filters.

1. INTRODUCTION

More complete mobile robot control architectures link task planning and reactive control. The task planning can compensate for the deficiencies of the reactive control and vice-versa (Rosenblatt, 1997; Arkin and Balch, 1997). The result is the hybrid architecture, including a reactive system to accomplish low-level tasks (e. g. to avoid obstacles) and a deliberative system to accomplish high-level tasks (e. g. to plan a path). However, this kind of control architecture is in general computationally too intensive and/or dependent of a “model of the world” for accomplishing the path planning. Because of this, they are generally not compatible with the limited computational resources onboard mobile robots, as well as with their dynamical working environment.

This work proposes a control architecture that consists in fusing the output of distinct controllers through a decentralised information filter (DIF). Although still being a behaviour-based architecture, it is flexible enough to allow some path planning in addition to reactive control depending on the controllers implemented, as it will become clear in this paper.

2. RELATED WORKS

There are some recent works in the literature that proposes a kind of behaviour fusion, like it is proposed in this paper. In the sequence of this section, five of these works are briefly characterised.

2.1 The AuRA Architecture (Arkin and Balch, 1997)

The AuRA (*Autonomous Robot Architecture*) architecture is a hybrid architecture composed by a hierarchical system responsible for the deliberative planning and a reactive system, known as S*cheme Controller*. It is a reactive system having a different control of its antecedents, like the subsumption architecture of Brooks (1986). The differences are that it is not necessary to arbitrate between distinct behaviours and there is no behaviour-layer. For each context of the working environment, the deliberative system defines which of the available behaviours should be used to accomplish the desired task. The schemes are codified using an analogy with the method of potential fields (Latombe, 1991). However, these schemes are much simpler than the classic method of potential fields in terms of computation. This is because the potential field in the AuRA architecture is calculated only in the current robot position and not in all points of the area surrounding the robot (Arkin, 1998). This allows the real-time execution of the schemes, which is basic for a reactive system. However, the methods based in potential fields are known to exhibit local minima and cyclic problems (Rosenblatt, 1997; Arkin and Balch, 1997; Arkin, 1998). This is one of the reasons to integrate deliberative planning to reactive control in the AuRA architecture.

A sensor scheme is associated to each motor scheme. The perception is structured in the sense of *knowing what is necessary*: every sensor scheme provides the necessary information to the reaction of the behaviours. The response of each motor scheme to the received excitation occurs as vectors. The motor schemes can operate in an asynchronous way, thus generating faster responses. To accomplish a task, some behaviours are grouped in a suitable way and the relative importance of each of them is coded through a gain. The vectors produced as the output of each behaviour (or scheme) are multiplied by the gains associated to them and a vector addition is performed to obtain the overall reaction of the robot to the working environment. The gains can be changed by a planning system or by a human operator, according to the task to be accomplished.

Thus, AuRA is an asynchronous architecture presenting continuous response codification, using an analogy to the potential fields. The method for coordinating the distinct behaviours is a co-operative one, through weighted vector addition (Arkin, 1998).

2.2 The DAMN Architecture (Rosenblatt, 1997)

DAMN (*Distributed Architecture for Mobile Navigation*) is an architecture in which various behaviours operating independently and asynchronously determine the robot action in a co-operative way. It consists of a group of asynchronous and distributed behaviours that send votes for the actions that satisfy their objectives and against those not satisfying them to a central arbitration system. Thus, they indicate to the arbitration system the usefulness of possible "world states". The arbitration system, then, is responsible for executing the fusion of the votes coming from the behaviours and for generating actions reflecting the current system objectives as well as possible. A weight is associated to all the behaviours, thus allowing establishing a priority scheme.

It may exist more than one arbitration system working in parallel to control the linear and angular speeds of the robot. This is a particular characteristic of the DAMN architecture (Arkin, 1998). The use of shared and distributed control allows multiple planning levels to be used in a decision-making without needing a hierarchical structure. Besides this, the distributed and asynchronous nature of the architecture allows taking into account distinct objectives and limitations simultaneously. Thus, the DAMN architecture provides objective-oriented behaviours that are rational and coherent, besides preserving the capability of real-time response.

In a few words, DAMN is an asynchronous architecture presenting discrete response codification, through sets of votes, and a method of behaviour coordination consisting in multiple independent *winner-takes-all* arbitration systems (Arkin, 1998).

2.3 Dynamical Approach (Bicho, 1999)

The dynamical approach includes concepts and principles that are based on the mathematical theory of dynamic systems and on neural networks. Such concepts are used as a theoretical language in developing control architectures for mobile robot navigation. The main ideas are:

1. the concept of *behavioural variables*, which are variables that can internally describe, represent and parameterise a certain behaviour (system state). Such variables define the domain in which the behaviour should remain in, and are chosen such that in each instant a behaviour is associated to particular values of its behavioural variables and the task requirements are expressed as values or set of values of these variables;
2. the concept of *behavioural dynamics*, according to which the behaviours are generated as attractive solutions of dynamic systems;
3. the dynamics of neural fields extends the former principles to the concept of neural representation of the information.

The time course of the behavioural variables is obtained as solutions of attractors for dynamic systems (behavioural dynamics), formulated to express the system requisites, as attractive or repulsive forces. Through the choice of the variables and the adjustment of their time courses, the system should be tuned to be always in an attractor, or at least close to one. Information coming from the sensors or from other behavioural modules determines the localisation, force and range of the attractive or repulsive contribution of the behavioural dynamics. Multiple contributions to the behavioural dynamics, like the sources of sensorial information, may co-operate or compete, thus resulting in a great behaviour change or just in a behavioural adjustment.

The dynamic approach applied to the control of mobile robot navigation, in a few words, is a synchronous architecture presenting continuous response codification (through vector fields). In addition, its behaviour co-ordination consists in the addition of the vector fields produced by each one of them.

2.4. *Multivaluated Logic Approach* (Saffiotti *et al.*, 1995)

Three layers compose this architecture: *control schemas*, *behaviour schemas* and *planners*. They interact with the robot and the environment in distinct levels of abstraction: signal/stimulus, intermediary (symbol to signal) and symbolic, respectively.

Control schemas describe types of movement based on the internal state of the robot and on the data coming from the sensors. The planners, on the other extreme, synthesise plans that are based in symbolic descriptions. The behaviour schemas fill the gap between the planners and the control schemas, containing parts in the level of either symbolic abstraction or stimulus.

The control schemas are defined as a mapping of a set S of states to preferences in a set A of action to execute. Formally, the mapping $D: S{\times}A \rightarrow [0,1]$, measures the degree of desire $D(s,a)$ of executing the action a in the situation s. In practice, each control schema is implemented by a set of IF-THEN fuzzy rules. The outputs of these rules are then combined in a membership function that reflects the degree of desire D of the control schema implemented. The membership function is *defuzzyfied* to calculate a single control output. Various control schemas can be composed using fuzzy logic operators, like OR, AND, etc.

Although the *defuzzification* process produces satisfactory results most times, in other situations, like when there is a conflict between two control schemas, it may produce non-suitable results. To avoid this problem, this approach proposes to associate a context to each control schema to define the conditions for a controller to be considered. The contextual conditions are implemented by a set of meta rules which activates the control schemas, like:

IF *context* = *A* THEN *activate_control_schema C*,

where A is a fuzzy term representing the real context. This fuzzy rule activates the control schema C in a level determined by how true is the rule antecedent. The activation level is used to weight the function degree of desire of the control schema.

Behaviour schemas are structures that connect certain standards of actions, implemented by control schemas, to certain stimulus coming from the environment. This way, stimulus-response behaviours can be formulated to meet specific objectives. Behaviours are described by $B = (C,D,O)$, where D is a control schema for a specific type of movement, O is a set of descriptors of an object with respect to the movement should be executed, and C is a context defining the applicability of the movement. Object descriptors are models of objects of the real world and assure that the behaviour acts over the external world, in opposition to a control schema, which acts according to internal variables. The behaviour schemas can also be integrated to form more complex behaviour schemas by using the same technique used for the control schemas.

Planners (based on the classical Artificial Intelligence theory) can use specifications of the behaviours, in terms of pre- and post- conditions, to build plans that allow accomplishing specific objectives.

Thus, this is a synchronous architecture presenting discrete response codification and a method for co-ordinating distinct behaviours called context-dependent bleeding.

2.5 *Multiple Objective Decision Making Control (Pirjanian, 2000)*

In this control architecture, all behaviours calculate an objective function for the admissible set of control actions. The action that maximises the objective function corresponds to the action that better satisfies the objective of the behaviour. The multiple behaviour are mixed in a single more complex behaviour that selects the action that simultaneously satisfies the objectives of each behaviour in the best way, which is a method of vector optimisation.

Thus, this is a synchronous architecture presenting discrete response codification and a behaviour co-ordination based on the theory of multiple objective decision-making (Chankong and Haimes, 1983).

3. THE PROPOSED CONTROL ARCHITECTURE

The proposed control architecture is based in the fusion of the output of a set of controllers by using a decentralised information filter (DIF). The set of controllers to be used depends on the specific application. Figure 1 represents an implementation of the proposed control architecture for a robot navigating inside an office building. As shown in the figure, each controller receives sensorial information and produces linear/angular velocities as its output, which are inputted to the local information filters. These local filters, together with a global information filter, are referred to as the decentralised information filter (Freire, *et al.*, 2001). Notice that the sensorial information each controller receives can be fused either for a better environment representation or for reducing the measurement noise.

In a DIF, a covariance measuring the confidence of the observed data is associated to each local filter. For the proposed control architecture, the interpretation and the calculation of these covariances are presented in Section 4. The output of the global information filter is closer to the output of the local information filter associated to the lowest covariance (the more reliable output). This way, the system combines information on the angular and linear velocities coming from different controllers using the DIF, which is an optimised fusion method (Mutambara, 1998), in opposition to the control architectures outlined in Section 2. Besides, it performs the fusion directly on the linear and angular velocities generated by each controller, thus not demanding any pre-fusion processing.

In a few words, the proposed architecture is a synchronous one presenting continuous response codification for the linear and angular velocities and using the fusion of control outputs through a DIF as the method for controllers co-ordination.

4. USING FUZZY LOGIC TO DETERMINE THE COVARIANCE OF THE DIF

When using an information filter (Mutambara, 1998), the covariance associated to the measurement error is a statistical measure of the confidence of the data provided by each information source (sensors). When dealing with sensors (sensor fusion), the covariance matrix is obtained by testing each sensor involved in the fusion process.

In the case of fusing the output of different controllers, the covariance represents a measure of how suitable a certain controller is regarding the current environmental condition. The lower the covariance associated to a certain controller is, the more suitable it is. Now, supposing that a reliable inference of environmental conditions is available, the designer can associate a suitability degree to each controller in each instant. This means to associate the corresponding covariance to each controller. The environmental conditions necessary to evaluate the suitability degree of each controller are inferred from the information either of the sensing system or provided by a supervisory system. In the former case sensor fusion can be used, and in the last one the supervisor can or can not have information on the structure of the working-environment.

The problem of statistically modelling either the failures of the sensors or the noise included in the measurements is too complex, mainly when the mobile robot is designed to operate in various environments. Such working environments are normally not structured or able to be adapted to the navigation conditions. Besides this, it is desirable that the mobile robot can operate in environments where some people evolve inside, and the action of such people can not be described by a deterministic or even stochastic process (Saffiotti, *et al.*, 1995). Because of this, statistical models of the sensors are too difficult to be obtained (Pirjanian, *et al.*, 1998) and are not considered in this paper. However, the system designer can include information about the confidence of a certain sensor when associating a covariance to a controller. Suppose, for instance, that the robot has two controllers for navigating in a corridor. One of them is based on the information provided by ultrasonic sensors and the other one is based on the difference between the optical flow measured on the walls in the left and the right side of the corridor. The ultrasonic sensors can

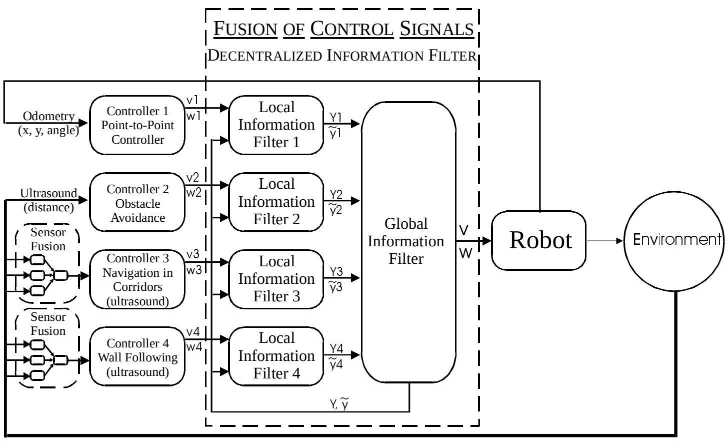

Fig. 1: The proposed architecture, which performs the fusion of the output of distinct controllers.

fail for several reasons, such as multiple reflections or even for cross talking. In addition, they have low angular resolution. By its turn, the optical flow can be misinterpreted due to either inappropriate illumination or poor wall texture. In this case, knowing the wall type and the illumination of the robot working-environment, the designer can assign a lower covariance (more importance) to the controller based on the information coming from the sensor better adapted to the environmental condition.

One way to determine the degree of suitability of each controller is to use mathematical relationships involving the data measured by the robot sensors. The more suitable a certain controller is, the lower should be the result of the equation defining its covariance. An easier way to calculate the covariance is the use of fuzzy logic. In this case, some linguistic variables are used to model the designer's knowledge about the robot navigation system and the robot working-environment through a rule base. This is the solution adopted in this work.

To determine the covariance using fuzzy logic, three fuzzy variables (antecedents) are used here. They are the least distance measured by the frontal ultrasonic sensors (d_{min}), the product of the distances measured by the ultrasonic sensors at the right (d_r) and left (d_l) sides and the least value between d_l and d_r. The fuzzy sets and the membership functions of these antecedents are identical and are shown in Figure 2. The consequent is the covariance associated to each one of the four controllers. They are R1 for the controller responsible for the navigation in corridors, R2 for the point-to-point controller, R3 for the controller responsible for obstacle avoidance and R4 for the controller responsible for wall following. They are modelled as singletons, which means that they are modelled as real numbers (Babuska, 1998). The defuzzyfication is performed by using the fuzzy mean method, which means that

$$y = \frac{\sum_{i=1}^{k} \beta_i b_i}{\sum_{i=1}^{k} \beta_i} \qquad (1)$$

where y is the value to be assigned to the covariance, b_i are the singleton values and β_i is the degree of pertinence of the antecedent. Tables 1 and 2 show the fuzzy rules adopted to determine the covariance assigned to each controller. There, TS means too small,

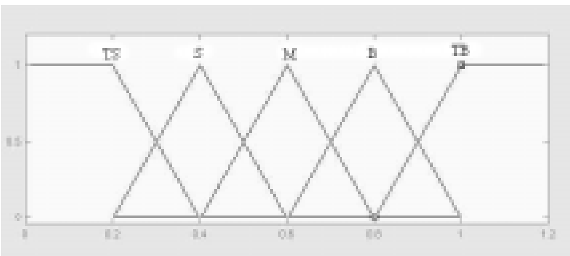

Fig. 2: Membership function of the input variables.

Table 1 Fuzzy rules determining the covariance of the controller responsible for corridor navigation

$d_l \times d_r$ / d_{min}	TS	S	M	B	TB
TS	R_1=TB	R_1=TB	R_1=TB	R_1=TB	R_1=TB
S	R_1=S	R_1=M	R_1=B	R_1=TB	R_1=TB
M	R_1=TS	R_1=TS	R_1=TS	R_1=TB	R_1=TB
B	R_1=TS	R_1=TS	R_1=TS	R_1=TB	R_1=TB
TB	R_1=TS	R_1=TS	R_1=TS	R_1=TB	R_1=TB

Table 2 Fuzzy rules used to determine the covariance of the point-to-point, obstacle avoidance and wall following controllers.

Min(d_l,d_r) / d_{min}	TS	S	M	B	TB
TS	R_2=TB	R_2=TB	R_2=TB	R_2=TB	R_2=TB
	R_3=TS	R_3=TS	R_3=TS	R_3=TS	R_3=TS
	R_4=TB	R_4=TB	R_4=TB	R_4=TB	R_4=TB
S	R_2=TB	R_2=TB	R_2=TB	R_2=TB	R_2=TB
	R_3=TS	R_3=TS	R_3=TS	R_3=TS	R_3=TS
	R_4=S	R_4=M	R_4=B	R_4=TB	R_4=TB
M	R_2=TB	R_2=TB	R_2=TB	R_2=TB	R_2=TB
	R_3=TB	R_3=TB	R_3=TB	R_3=TB	R_3=TB
	R_4=TS	R_4=TS	R_4=TS	R_4=TB	R_4=TB
B	R_2=TB	R_2=TB	R_2=B	R_2=S	R_2=TS
	R_3=TB	R_3=TB	R_3=TB	R_3=TB	R_3=TB
	R_4=TS	R_4=TS	R_4=TS	R_4=TB	R_4=TB
TB	R_2=TB	R_2=TB	R_2=B	R_2=S	R_2=TS
	R_3=TB	R_3=TB	R_3=TB	R_3=TB	R_3=TB
	R_4=TS	R_4=TS	R_4=TS	R_4=TB	R_4=TB

S means small, M means medium, B means big and TB means too big. These rules can be interpreted in the following way. In the case d_{min} is small, the collision risk is big, so it must be assigned a small covariance to the obstacle avoidance controller, while a bigger one must be assigned to the other controllers. When d_{min} is not small, the covariance assigned to the obstacle avoidance controller should have a big relative value. If besides this, the product $d_l \times d_r$ is small, it means that the robot is evolving in a corridor. So, the corridor navigation controller is the one that must have the smallest covariance assigned to it. If d_{min} and the product $d_l \times d_r$ are not small but d_l or d_r is small, there is a wall in the left or right side of the robot. In this case, the wall following controller is the one to which it must be assigned the smallest covariance. Finally, case d_{min}, d_l, d_r and the product $d_l \times d_r$ are not small, the robot should be in an open environment. Thus, the point-to-point controller should have the smallest covariance.

5. EXPERIMENTAL RESULTS

To demonstrate the performance of the proposed approach, a practical navigation experiment consisting in guiding the robot from an initial point [0,0] to a destination point [12m,5m] in an office building environment was considered. While evolving towards the destination point, the robot should navigate through corridors and avoid colliding with obstacles.

The experiment was run using a PIONEER 2 DX mobile robot, which has sixteen ultrasonic sensors (only ten were used). The navigation is controlled from an onboard 500 MHz K6-II computer running the proposed control architecture. Figure 3 shows the path followed by the robot, and demonstrates its capability to successfully reach the destination point in a relatively complex unknown environment.

To evaluate the performance of the control system, four indexes have been considered (Pirjanian, 2000). Table 3 shows the resulting values for the performance indexes, including their ideal values. The *safety* index indicates the minimal distance measured by the ultrasonic sensors along the robot path (it indicates the risk of collision). Then, the robot navigated in a safe way during the experiment. The *average speed* (linear speed) index indicates the average linear speed along the robot path. As one can see, the fusion of distinct control signals effectively makes the robot to navigate a little slower. Finally, the *smoothness* index is calculated as the average magnitude of the difference between the current and the previous robot orientation, thus showing how smoothly the manoeuvres are performed. As one can see, the proposed architecture allows very smooth manoeuvres.

CONCLUSION

A new control architecture is proposed to control the navigation of a mobile robot. It consists in the fusion of the outputs of different controllers to generate the overall control signal. The proposed fusion technique is optimal due to the use of a decentralised information filter (DIF). It is also proposed to use fuzzy logic to define the covariance associated to each local information filter.

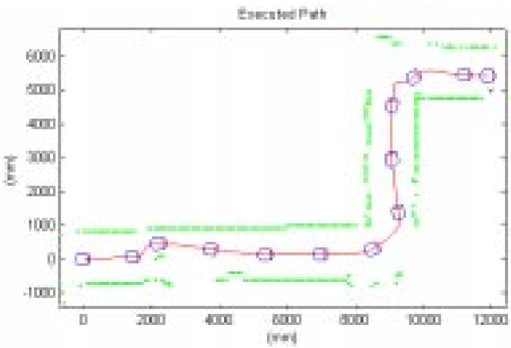

Fig. 3. The path followed by the robot when using

Table 3 Performance evaluation indexes

Index	Obtained value	Ideal value
Safety	212 mm	500 mm
Average velocity	273 mm/s	300 mm/s
Smoothness	0.86^o	0
Travelled distance	16.91m	17.00 m
Elapsed time	57.3s	56.7s

Compared to the other control architectures here addressed, the proposed one uses the DIF as a fusion approach, which guarantees an optimised result. It also presents the advantage of directly realising the fusion of the control outputs (linear and angular velocities), thus not demanding any data pre-processing. Finally, it allows a mathematical analysis of control stability, which is now being accomplished.

ACKNOWLEDGEMENT

The authors thank CAPES, FAP-SE and ITP-UNIT (Program to Support Institutional Research), from Brazil, and SETCIP, ANPCyT and CONICET, from Argentina, for granting this project. In special, they thank CAPES and SETCIP for supporting a partnership between Universidade Federal do Espírito Santo/BRAZIL and Universidad Nacional de San Juan/ARGENTINA.

REFERENCES

Arkin, R. and T. Balch (1997). AuRA: principles and practice in review. *Experimental and Theoretical Artificial Intelligence*, **9**, 175-189.

Arkin, R. C. (1998). *Behavior-based robotics*. The MIT Press, Reading, MA, USA.

Babuska, R. (1998). *Fuzzy modelling for control*. Kluwer Academic Publishers, USA.

Bicho, E. (1999). *The dynamic approach to behavior-based robotics*. PhD. Thesis, University of Minho, Portugal.

Brooks, R. A. (1986). *Achieving artificial intelligence through building robots*. MIT, A.I. Memo 899, Reading, MA, USA.

Chankong, V. and Haimes, Y. (1983). *Multiobjective decision-making – theory and methodology, Vol. 8*, North-Holland, Amsterdan, Holland.

Freire, E. O., Carelli, R., Mut, V., Soria, C., Bastos Filho, T. F. and Sarcinelli-Filho, M. (2001). Mobile robot navigation based on the fusion of control signals from different controllers. In: *Proceedings of the 2001 European Control Conference*, 1828-1833. Porto, Portugal.

Latombe, J. C. (1991). *Robot motion planning*. Kluwer Academic Publishers, Boston, USA.

Mutambara, A. G. O. (1998). *Decentralized estimation and control for multi-sensor systems*. CRC Press, USA.

Pirjanian, P., Christensen, H. and Fayman, J. (1998). Application of voting to fusion of purposive modules:an experimental investigation. *Robotics & Autonomous Systems*, **23**, 253-266.

Pirjanian, P. (2000). Multiple objective behaviour-based control. *Robotics and Autonomous Systems*, **31**, 53-60.

Rosenblatt, J. (1997). *DAMN: A distributed architecture for mobile navigation*. PhD Thesis, Carnegie Melon University, USA.

Saffiotti, A., K. Konolige and E. Ruspini (1995). A multi-valuated logic approach to integrating planning and control. *Artificial Intelligence*, **76**, 481-526.