Inteligência Artificial

Construção Civil
Desperdício de material

Caderno de Artigos Técnicos
AN EXTERNAL SENSING SYSTEM FOR A MOBILE ROBOT

Eduardo Oliveira Freire
Teodiano Freire Bastos
Hansjörg Schneebeli

Department of Electrical Engineering, Federal University of Espirito Santo, CP 01-9011,
Vitória - ES, 29060-970 - Brazil - Fax: +55-27-335-2650

Resumo: Um robô móvel está sendo desenvolvido no Departamento de Engenharia Elétrica
da Universidade Federal do Espirito Santo. O robô possui uma arquitetura de controle
baseada em agentes e realizará o sensoriamento externo com sensores de ultra-som e um
sistema de visão artificial. Este sensoriamento será capaz de fornecer a informação sobre a
distância e o tipo de obstáculo detectado.

Palavras-Chave: Robôs Móveis; Sensores; Transdutores Ultra-Sônicos; Visão Artificial; Agentes

Abstract: The availability of sensorial data is a major drawback for mobile robot navigation
systems. For a mobile robot under construction at the Department of Electrical Engineering
of the Federal University of Espirito Santo, Brazil, an agent-based control system is being
implemented, for which ultrasonic and vision systems are being developed. This sensing
system is able to provide the information on the type of detected obstacle as well as on the
distance from it to the robot.

Keywords: Mobile Robots; Sensors; Ultrasonic Transducers; Computer Vision; Agents.

1. INTRODUCTION

A differential drive robot using DC motors is under
construction at the Electrical Engineering Department
(DEL) of the Federal University of Espirito Santo
(UFES), Brazil. This mobile robot is a round
platform with four wheels; two of them are mounted
on a common axis and are independently driven and
the two other are free wheels. Figure 1 shows that
mobile robot, called Brutus. It is possible to see,
along with the round platform outer limit, some of the
ultrasonic transducers. The round platform and the
differential drive construction allow the robot to
easily steer.

This mobile robot uses an MC-68332 microcontroller
to control the several processes involved in the
mobile robot movement (e.g. internal sensorial
perception and trajectory planning). The robot is
designed to be able to avoid any collision, when
moving itself to execute some task in a semi-
structured environment. This is possible due to the
ultrasonic sensing system developed.

Semi-structured environments are those in which the
limits are not perfectly known by the system, but that
have closed configuration and regular ground. The
robot operating environment should have a finite
number of known objects (walls, corners, chairs and
tables), even though unknown objects can suddenly
appear in the trajectory of the robot.

Figure 1: The mobile robot Brutus.
2. AN STRUCTURE FOR CONSTRUCTING AGENT-BASED CONTROL SYSTEMS

Schneebeeli (1992) presented an abstraction for agent-based controllers, in which they are represented by concurrent and interconnected modules, distributed in three categories: sensor agents (primitive sensor agents and virtual sensor agents), behavior agents and actuator agents (virtual actuator agents and primitive actuator agents). They are shown in Figure 2. In this approach, like in Brooks' (Brooks, 1986), a group of modules are responsible for an activity (behavior) to be executed by the robot. Each activity represents a system goal. The interaction and mediation between various modules define the main goal.

This module categorization provides an abstraction that makes the purpose of each module closer to its representation, giving to each agent category some particular characteristics, related to status and communication, and associated to their specific role. For example: the actuator agents have the characteristic of being commanded by just one behavior agent at any time. The sensor drivers are represented by primitive sensor agents while tasks like build maps are represented by virtual sensor agents. The modules that define actions or behaviors are represented by behavior agents and tasks like move forward are represented by virtual actuator agents. Finally, the actuators output drivers are represented by primitive actuator agents.

The structure for constructing agent-based control systems developed by Xavier and Schneebeeli (1995) implements Schneebeeli's (1992) abstractions using the C++ programming language and concurrent libraries. So, a concurrent object-oriented structure is built. This structure is very suitable for behavior-based systems modeling because it joins characteristics such as modularity, encapsulation, inheritance, concurrence and the C++ efficiency.

In the structure aforementioned, the control modules (agents) are constructed as instances of user defined classes derived from the sensor, behavior and actuator classes. So the definition of just a class can be used to create multiple agents, by instantiation and, if necessary, at running time. In such case, the difference between the created agents would be given by the arguments passed to the constructor of the class, and by the communication gates established.

The dynamic instantiation and desinstantiation of agents plus the dynamic inter-module connection mechanisms defined in the structure (Yonesawa, 1988), provide a dynamic reconfiguration capability to the system that can be used to help the robot to handle and manage diverse situations using different control strategies.

Mechanisms of inheritance and the modularity provided by the structure ensure the easy expansion of the control system. So the creation of new kinds of agents can be done by deriving classes from the sensor, behavior and actuator classes, thus minimizing code rewriting. By its turn, the data and functions encapsulation in classes and objects ensures the system modularity.

3. RANGE MEASUREMENT AND OBJECT RECOGNITION TECHNIQUES

Attached to this agent-based control structure, it was developed a software for controlling and testing the ultrasonic sensing system. This software was written using the Borland C++ 3.1 programming language. In this section, the algorithms there implemented are discussed.

3.1. Range Measurement Technique

The technique here used to perform the range measurement is the well known Pulse-Echo Method. Using this technique, the distance between the sensor and the detected obstacle is given by:

$$ d = \frac{c}{2} t $$

where $c$ is the velocity of sound in the air and $t$ is the time of flight (TOF) associated to the echo ultrasonic signal.

Figure 3 shows the performance of the implemented ultrasonic sensing system in range measurement. The experiment was done using a wall (infinite plane) at a distance of 50cm from the ultrasonic sensor. The mean distance value obtained was 50.09 cm and the standard deviation was 9.14 cm. Even though they are not so precise, the obtained results are very good for mobile robot navigation applications, mainly when considering the very simple structure implemented to get them.
3.2. Object Recognition Technique

In order to obtain more information about the operating environment of the robot, an algorithm was implemented to allow the ultrasonic sensing system to recognize the detected obstacle. The system is able to recognize the following types of obstacles: walls and cupboards (planes), corners, table feet (rectangular and cylindrical reflexive areas) and chair feet. This ability makes the robot control task more flexible, since it is possible, for example, to distinguish a wall from a table foot.

The technique used consists in measuring the peak value of the echo signal reflected by the detected obstacle and comparing it with the value furnished by the following equations, which relate the peak value of the echo signal to the distance from the ultrasonic transducer to each type of obstacle (Abreu, 1990):

\[ U_p(x) = \frac{K_p e^{-\alpha x}}{x} \]  \hspace{1cm} (2)

\[ U_c(x) = \frac{K_c e^{-\alpha x}}{x^2} a \cdot b \]  \hspace{1cm} (3)

\[ U_{cf}(x) = \frac{K_{cf} e^{-\alpha x}}{x} \sqrt{\frac{r}{x+r}} \]  \hspace{1cm} (4)

In Equations (2) to (4), \( K_p, K_c \) and \( K_{cf} \) are constants, \( x \) is the distance from the ultrasonic transducer to the detected obstacle, \( \alpha \) is the attenuation coefficient of the air (equal to \( 1.61 \times 10^{-15} \) \( s^2/cm \)), and \( r \) is the cylinder radius (associated to cylindrical table feet).

Figure 4 shows the experimental and theoretical curves relating the echo signal peak value and the distance, regarding the distance interval in which the obstacles can be recognized (Freire 1997). Equation (2) is used to compute that relationship for planes (walls and cupboards), whereas Equation (3) is used for rectangular table feet and Equation (4) is used for cylindrical table and chair feet.

As shown in Figure 4, it is possible to recognize all the obstacles, except for the corners and cylindrical table feet, whose recognition fails because of possible misinterpretations when comparing the echo signal peak values. This remaining problem was solved by using a very particular building characteristic of the ultrasonic sensing system, what allows the corner recognition to be performed as shown in Figure 5. Then, the corners are no longer recognized using the echo signal peak value, thus eliminating the misinterpretation when defining if the detected obstacle is a corner or a cylindrical table foot.

4. A STRATEGY FOR ULTRASONIC SENSORS ACTIVATION

In the agent-based architectures already developed, the sensor agents often are very simple: they fire behaviors when their outputs exceed a threshold. So, the sensor agents are often periodically started. This characteristic limits the sensing system performance.

This paper proposes a non-periodic way of starting the sensor agents. In order to do this, an Object Oriented Concurrent Structure developed by Xavier and Schneckewell (1995) for building agent-based systems is used. The transducers are fired according to the priority associated to each one. When the priority increases, the firing frequency also increases.

Figure 5: Recognizing corners (Freire, 1997).

The priority assigned to each transducer \( pT(x) \) varies according to the function:

\[ pT(x) = \frac{1}{1 + e^{-\alpha x}} \]
where \( k \) is the \( k \)th transducer, \( x \) is the distance from the transducer to the detected obstacle; \( c \) is the minimum priority assigned to each sensor agent (\( c > 1 \)); \( \alpha \) is the angular displacement of the \( k \)th transducer related to the reference transducer; \( n = 0 \) means that the robot front is in the same direction of the reference transducer (T1); \( n = 1 \) means that the robot front is in the opposite direction (the value of \( n \) is determined by a virtual actuator agent named \texttt{Activate\_motor}), and \( a \) and \( b \) are adjustable constants.

The first term in Equation (5) means that the priority of the sensor agent is greater when it detects an obstacle that is closer to it. The second term guarantees that the priority of one sensor agent increases as its associated transducer becomes closer to the detected obstacle. The last term (a constant which value varies according to \( \alpha \) and the direction of the robot displacement), by its turn, avoids the assignment of very small priorities to the sensor agents. The concept of "small", varies according to the associated transducer location around the robot platform.

5. USING AN ARTIFICIAL VISION SYSTEM

The use of ultrasonic sensors as the unique source that provides the information to the robot navigation task is not safe enough. The problem is that the ultrasonic sensors have limited operation and they suffer influences of several environment factors, such as temperature changes and air turbulence, for example (Bastos, 1994).

Thus, the object recognition technique described in Section 3.2 is very useful, by its simplicity, but it is not too safe. So, an artificial vision system is normally used for object recognition (Brooks, 1985; Horn, 1989), in order to get more information of the operating environment, thus insuring the safety of the robot navigation and allowing the robot to execute more complex tasks. In this work, a vision system is used, but it will be fired only when the ultrasonic system is not able to recognize the detected object. This is also an strategy for minimizing onboard computation, what is a serious drawback when using vision systems as external sensing systems.

Before implementing the routines for onboard image processing, several algorithms are being studied in a computer running Linux, which is connected to a MicroSys computer through a serial port. The reason for using the MicroSys computer is that it is based on a Motorola 68000 microprocessor which is the same family of another 68332 microcontroller to be used to process the images acquired by a monochromatic camera (SONY model XC-77). The software "Minicom" is responsible for the communication between both computers. The software developed for MicroSys computer is generated using a GCC cross-compiler running under Linux. The camera is connected to an image processing board (model ED-68VIN-4) installed in the MicroSys computer. All the software are being developed in C++. The system to carry out the image processing is shown in Figure 6.

Once the steps aforementioned be completed, the object recognition task is carried out in the following way: the camera is mounted on a stepping motor driven system located on the top of the robot platform (Figure 7), such that when one of the ultrasonic sensors detects an obstacle closer to the robot than a certain threshold distance (about 1m), the control system sends an order to rotate the camera towards the ultrasonic sensor that detected the obstacle.

6. AN EXAMPLE OF AGENT-BASED CONTROL SYSTEM

In order to show how the external sensing system developed can be joined to the central control unit, an example of an agent-based control system that is able to use the information provided by the ultrasonic sensing system is shown in Figure 8.
Figure 8: Example of an agent-based control system.

This control system is composed by 16 primitive sensor agents called \textit{S}, two primitive sensor agents called \textit{Encoder1} and \textit{Encoder2}, one primitive sensor agent called \textit{Cam_Motor}, two virtual sensor agents, one called \textit{Ultrasonic_sensor} and other called \textit{Camera_Sensor}, four behavior agents, called \textit{Collision_avoidance}, \textit{Robot_align}, \textit{Recognizer} and \textit{Map_builder}, and finally three virtual actuator agents and one primitive actuator agent, called \textit{Go_ahead}, \textit{Rotate}, \textit{Activate_motor} and \textit{PID} (Proportional Integral and Derivative controller) agent, respectively.

These agents are connected in order to implement four behaviors or control levels, named \textit{Motor Control, Explore, Recognize Obstacles and Build Map}.

The \textit{S} agents are responsible for firing the ultrasonic transducers and processing the echo signal (range measurement, envelope digitalization, echo signal peak value calculation and for calculating the area under the first echo). These information, besides the absolute priority associated to the \textit{S} agent (calculated through Equation (5)), are then passed as a message to the \textit{Ultrasonic_sensor} agent, that keeps two lists, one with the distance measured and other with the absolute priority assigned to each \textit{S} agent. When the \textit{Ultrasonic_sensor} receives a message that contains a distance lower the 100cm (indicating that an obstacle at this distance was detected), it sends a message activating the \textit{Collision_avoidance}, \textit{Robot_align} and \textit{Camera_Sensor} agents.

When each \textit{S} agent sends a message to the \textit{Ultrasonic_sensor}, it receives from this agent a message informing the sum of the absolute priority assigned to each \textit{S} agent. So, each \textit{S} agent is able to calculate its relative priority by dividing its absolute priority by the sum received from the \textit{Ultrasonic_sensor} agent. The inverse of its relative priority is used to program its internal watch, thus changing the interval of time in which the agent is activated. As the hardware resources are shared by all ultrasonic sensors, when an \textit{S} agent is activated it goes to a queue and wait its time. As the priority assigned to an \textit{S} agent grows, the interval of activation reduces and the \textit{S} agent goes to the queue more frequently.

If the \textit{Ultrasonic_sensor} agent receives a message that contains a distance measurement lower than 40cm, the \textit{Robot_align} and the \textit{Camera_Sensor} agents deactivate themselves and the \textit{Collision_avoidance} agent starts the procedure to avoid obstacles. Initially, the agent sends a pause message to the \textit{Go_ahead} agent, that stops the robot while it turns around its own central axis, due to a message sent by the \textit{Collision_avoidance} agent to the \textit{Rotate} agent. The rotate direction and velocity are randomly determined. After some time, measured by the object \textit{Watch} associated to the \textit{Collision_avoidance} agent, this agent sends a message deactivating the \textit{Rotate} agent and, then, the pause message sent to the \textit{Go_ahead} agent is suspended. Now the robot continues exploring its operating environment.

If the rotation was not enough to avoid the obstacle, certainly another \textit{S} agent will detect this obstacle at a distance lower than 40cm again and the process will be repeated. The \textit{Collision_avoidance} agent have a kind of entrance called \textit{canal} that may receive messages from all the \textit{S} agents and perform some operation on them, such as to calculate the mean value, the biggest value or the lowest value. In this case, the \textit{canal} informs to the \textit{Collision_avoidance} agent the lowest distance received (the direction of the most probable collision).

If the \textit{Ultrasonic_sensor} agent receives a message that contains a distance measurement in the range 40cm to 100cm, the \textit{Collision_avoidance} agent deactivate itself and the \textit{Robot_align} and \textit{Camera_Sensor} agents start the procedures to recognize the detected obstacle. Initially, the \textit{Robot_align} agent sends a pause message to the \textit{Go_ahead} agent, that stops the robot, and a message to the \textit{Rotate} agent, that rotates the robot around its own central axis looking for the best alignment between the ultrasonic transducer emission axis and the detected obstacle. When this orientation is reached, the \textit{Robot_align} agent calculates the mean values of the next ten distance measurements and area under the first echo, and sends these results in a message that activates the \textit{Recognizer} agent. If the recognition fails, the \textit{Camera_Sensor} agent starts the \textit{Cam_Motor} agent, which rotates the camera towards the \textit{S} agent that detected the obstacle. So, the \textit{Camera_Sensor} agent gets an image of the detected object and sends the results to the \textit{Recognizer} agent. This agent executes the recognizance of the detected...
obstacle with the information received from the Robot_align and Camera_Sensor agents, and sends the result in a message that activates the Map_builder agent. This agent is responsible for updating the map of the operating environment of the robot. Finally, a message sent by the Recognizer agent, after executing the object recognition procedure, suspends the pause massage sent to the GoAhead agent, thus allowing the robot to move again.

The Activate_motor agent also have a canal entrance that delivers to the agent the sum of the velocity components sent by the GoAhead and Rotate agents. The Activate_motor agent calculates the velocity that should be assigned to each DC motor and sends this information to the PID agent, which is responsible for controlling both motors.

7. CONCLUDING REMARKS

The ultrasonic sensing system built during the tests demonstrated to be able to provide the necessary information to allow the navigation of the robot on semi-structured environments.

The location of the ultrasonic transducers around the robot platform allowed the efficient and fast environment sensing, besides reducing the crosstalk problem, due to the fact that all ultrasonic sensors are aligned in different directions.

The strategy proposed for activating the ultrasonic sensors allowed the reduction of unnecessary or redundant information, thus making the robot navigation task easier and faster.

The ability to recognize obstacles, that became possible with the information provided by the ultrasonic sensors, will be improved with the artificial vision system (that insures more reliability). This will give more flexibility to the control system, which will allow the design of a greater number of behaviors, thus giving to the robot the possibility of executing more complex tasks.

Finally, the agent-based control system proposed shows how to join the external sensing system to the central control system.

8. ACKNOWLEDGMENT

We would like to thank to Professor Mário Sarcinelli Filho for helping us to correct the English transcription of this paper.

9. REFERENCES


