

## COMPARISON BETWEEN PID AND FUZZY CONTROLLERS USED IN MOBILE ROBOT CONTROL

POPESCU, C[ristina]; PARASCHIV, N[icolae] & CANGEA, O[tilia]

**Abstract:** Achieving the models of nonlinear systems is quite a difficult task and often simplified forms are obtained due to the simplifying assumptions used. The goal of this paper is to present an overview of fuzzy controllers used in mobile robots and to discuss the advantages and drawbacks of fuzzy control in comparison with PID control.

**Key words:** mobile robot, fuzzy control, PID control

### 1. INTRODUCTION

Opinions about fuzzy controllers are different: some authors consider that artificial intelligence, especially fuzzy control represent a revolution in intelligent control (Suster & Jadlovska, 2011), while others believe that what can be achieved using fuzzy methods can also be achieved using conventional techniques, like PID control (Pease, 1993).

In this paper the fuzzy controller is compared with a classical PID, because the PID is a reference controller. Both controllers will be used for differential mobile robot control, as Khepera III, studied by the authors in previous works. A PID controller has only three adjustable parameters. The controlled systems show good results in terms of response time and precision when these parameters are well adjusted. The fuzzy controller has many parameters and the most important requirement is to make a proper choice of the rule base, the scaling factors and the defuzzification method. We have observed that the fuzzy controlled system is very sensitive to the distribution of membership functions but not to their shape.

This paper want to highlight the advantages and disadvantages of using a fuzzy controller compared with a PID for nonlinear systems control, particularly of mobile robots. These results can be used for future research in automation of modelling process of fuzzy controllers.

### 2. CLASSICAL CONTROL APPROACH

The PID is proved to be a very efficient controller. The formula of the algorithm is:

$$u(t) = K_p e(t) + K_i \int_0^t e(t) dt + K_d \frac{de}{dt} \quad (1)$$

where  $u$  is the control variable,  $e$  is the error, defined as  $e=r-y$ ,  $r$  is the reference value and  $y$  is the process output. There are three parameters to adjust:  $K_p$ ,  $K_i$  and  $K_d$ .

For a differential mobile robot, such as Khepera III, it is necessary to obtain its kinematic model. This robot has two driving wheels driven independently with a common horizontal axis. Robot motion on straight trajectories is given by wheels constant velocity and the movement on circular trajectories is determined by the difference of velocity between the two wheels. The robot is localized by the characteristic point C located in the center of the wheel axis and the angle made by the perpendicular to the axis of the wheels with one of the external reference system axes ( $\theta$  direction), as illustrated in figure 1.

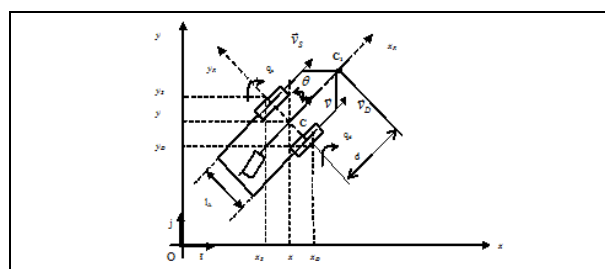


Fig. 1. Position and velocities parameters

Velocity of the robot is given by the point C velocity ( $\vec{v}$ ), and velocities of the two-wheel drive ( $\vec{v}_S$  and  $\vec{v}_D$ ) have the same direction perpendicular to the axis joining them. Robot state is defined by the position, orientation and velocities of the two driving wheels. In the essence the differential model of the robot is given by the following equations:

$$v = \frac{v_S + v_D}{2} \quad (2)$$

$$\omega = \frac{v_S - v_D}{L} \quad (3)$$

$$\dot{x} = \cos(\theta) \left( \frac{v_S + v_D}{2} \right) \quad (4)$$

$$\dot{y} = \sin(\theta) \left( \frac{v_S + v_D}{2} \right) \quad (5)$$

$$\dot{\theta} = \frac{v_S - v_D}{L} \quad (6)$$

As indicated in the Khepera user manual, both DC motors can be controlled by a PID controller executed in an interrupt routine of the main processor (K-Team, 2008). The motor controller can be used in two control modes: speed and position modes. The active control mode is set according to the kind of command received. If the controller receives a position control command, the control mode is automatically switched to the position mode. Different control parameters ( $K_p$ ,  $K_i$  and  $K_d$ ) can be set for each of the two control modes. The robot has two incremental encoders on the wheel. Using these sensors it is possible to measure the displacement and the speed of each wheel at every moment. Used in speed mode, the controller has as input the wheels speed value, and controls the motor to keep this wheel speed. No limitation in acceleration is considered in this mode. Used in position mode, the controller has as input a target position of the wheel, acceleration and a maximal speed. Using these values, the controller accelerates the wheel until the maximal speed is reached, and decelerates in order to reach the target position (K-Team, 2008). Both speed control system (figure 2) and position control system (figure 3) are classical PID control systems. With these control structures and an optimal setting of controller parameters it was observed a good behaviour for the robot requirements. The problem that emerges is that for a classical control of a mobile robot is required a precise knowledge of kinematic and dynamic model of it, that is difficult to obtain. For this reason are used increasingly more fuzzy controllers that not require the exact model of the system.

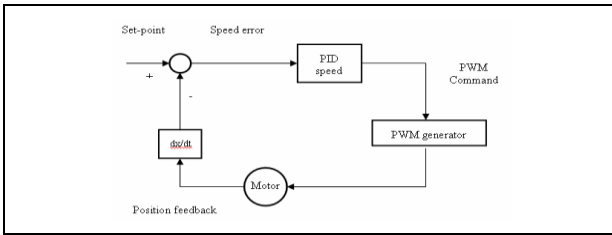


Fig. 2. Structure of PID speed control system

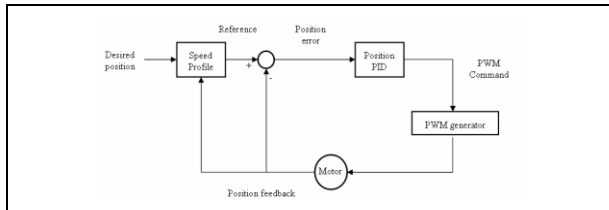


Fig. 3. Structure of PID position control system

**3. FUZZY CONTROL**

For accomplishing the many tasks of navigation, the robot must avoid obstacles in its way. For this reason, the experiment presented in this paper refers to the implementation of a simple controller for the Khepera mobile robot to obstacle avoidance behaviour.

The design of a fuzzy controller begins with the choice of linguistic variables, process states, input and output variables. The next step is the choice of the set of linguistic rules and the kind of fuzzy reasoning process. After the inference it has to be established a defuzzification strategy. The structure of a fuzzy controller is composed of four blocks: *the fuzzification interface* which performs the transformation of crisp inputs into fuzzy sets, *the knowledge base* which supplies the fuzzification module, the inference engine and the defuzzification interface with necessary information for their proper functioning, *the inference engine* that computes the meaning of the set of linguistic rules, *the defuzzification interface* that transforms the union of fuzzy sets into a crisp output (Godjevac, 1997).

To design a fuzzy controller for implementing obstacle avoiding behaviour it was supposed that we have all necessary needed information for robot navigation. The inputs are the linguistic variables: *distances between the robot and the obstacle*, and the outputs are the linguistic variables: *motor speeds*. Four input linguistic variables were used: *distance to the left  $D_s$ , distance to the right  $D_d$ , distance to the front  $D_f$  and distance to the back  $D_{sp}$* , as shown in figure 4. For each input linguistic variable, 3 linguistic values were defined: *small, medium, big*, while for each output we define 7 linguistic values: *backward fast, backward medium, backward slow, stop, forward slow, forward medium, forward fast*. The aim of the experiment is the implementation of a Sugeno fuzzy controller which allows navigation with obstacles avoiding behavior. For this controller, with 4 inputs and 2 outputs, the Matlab implementation offers a user interface shown in figure 5. Each membership function for input variables is Gaussian type while for output variables singleton functions were used.

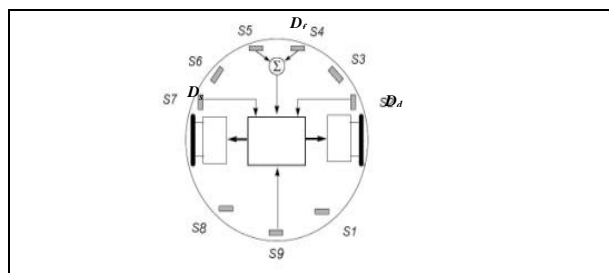


Fig. 4. Inputs variables of the robot  
The initial rules base describing obstacle avoiding behavior contains 16 rules



Fig. 5. Fuzzy controller with 4 inputs and 2 outputs

It was demonstrated that with 16 rules, Khepera avoid obstacles with success, but adding rules can lead to an inconsistent rules base or, worse, conflicts between rules can appear. For defuzzification it was used the weight average method and the outputs were calculated as follow

$$y_1 = \frac{\sum_{n=1}^N \bar{u}_n w_{n1}}{\sum_{n=1}^N u_n w_{n1}} \quad (7)$$

$$y_2 = \frac{\sum_{n=1}^N \bar{u}_n w_{n2}}{\sum_{n=1}^N u_n w_{n2}} \quad (8)$$

where  $y_1, y_2$  are the outputs controller,  $u_n$  are the firing strengths for the n rule and  $w_{n1}, w_{n2}$  are the parameters of n rule for the two outputs. With this rules base, the fuzzy controller was implemented with success on the real robot Khepera, the communication between the robot and the computer being made by Bluetooth technology (Popescu et al., 2009).

**4. CONCLUSION**

Standard controllers are suitable for systems that have an exactly defined mathematical model. In the design of a fuzzy controller it is not necessary to know a precise mathematical model of the plant to be controlled. An important problem of fuzzy controllers is that the computing time is much longer than of a PID, because of the complex operations: fuzzification, inference, and defuzzification. Some optimization can be obtained by simplifying the defuzzification method. *The main advantages of fuzzy controllers are:* the mathematical model of the plant is not needed; possibility to implement human expert knowledge and experience using linguistic rules; possibility to control non-linear plants. *Fuzzy controllers have some important drawbacks:* there is no general procedure for choosing the optimal number of rules, because of the many factors involved in the decision; the coherency of the rules is not guaranteed; knowledge of the human operator is often incomplete. As conclusion, the researchers have been trying to automate the modelling process of fuzzy controllers that could be divided in two parts: identification of the structure of a fuzzy controller and parameter identification (adjustment of the parameters of membership functions).

**5. REFERENCES**

Godjevac, J.(1997). *Neuro-Fuzzy Controllers Design and Application*, Presses Polytechniques et Universitaires Romandes, ISBN 2-88074-355-9, Lausanne

Pease, R.A.(1993). What's All This Fuzzy Logic Stuff, Anyhow?. Available from: <http://www.electronicdesign.com> Accessed: 2011-03-28

Popescu, C.; Paraschiv, N. & Cangea, O. (2009). Neuro-Fuzzy Controller for Mobile Robot Navigation with Avoiding Obstacles and Reaching target Behaviors, *Proceedings of the 20<sup>th</sup> International DAAAM Symposium*, pp. 0595-0596, ISBN 978-3-901509-70-4, Vienna, Austria, November 2009

Suster, P.; Jadlovska A.(2011). Tracking Trajectory of the Mobile Robot Khepera II Using Approaches of Artificial Intelligence, *Acta Electrotechnica et Informatica*, Vol.11, no.1 (2011), pp.38-43, ISSN 1338-3957

\*\*\* (2008) <http://k-team.com> – K-Team Corporation, Khepera User Manual, Accessed on: 2011-03-23