Evolutionary Optimization of the Fuzzy Integrator in a Navigation System for a Mobile Robot

Abraham Meléndez and Oscar Castillo

Tijuana Institute of Technology, Tijuana México
abraham.ms@gmail.com, ocastillo@tectijuana.mx

Abstract. This paper describes the optimization of an Integrator control block within the proposed navigation control system for a mobile robot. The control blocks that the integrator will combine are two Fuzzy Inference Systems (FIS) in charge of tracking and reaction respectively. The integrator block is call Weighted Fussy Inference System (WFIS), and assigns weights to the responses on each behavior block, to combine them into a single response.

1 Introduction

The use of mobile robots has increased over the last decades in many areas from industrial work to research and household and one reason for this is that they have proved useful in each of these areas from doing very specific task to ongoing monotonous shores, they help their human counterpart be more productive and efficient. Also as hardware technology is moving forward and developing more capable robots at lower cost, this is another reason for this increase and why we are seeing them in more common places.

The mobile robot, needs to move around its environment and this is why a great deal of research has been invested on testing them with control systems that allow the robots to navigate on their own, and different methodologies have been applied from traditional control such as PD, PID [4, 9] to soft computing methods like Fuzzy Logic [10, 25, 13, 17, 15, 18, 12, 11, 21, 5, 19, 20], Neural Networks [11] and hybrid ones also [6,23,24].

In this paper, the navigation control system has been designed to combine two key behaviors that are considered to be required for any navigation control system of a mobile robot. The first one is a tracking controller, this is an obvious one since there is no point of having a navigation system on a robot, that can´t go to a desired location, the second one is a reactive controller and here we considered this one to be off great importance also, since the tracking controller can get the robot to the destination, but that will be on an ideal situation where there are no obstacles present on the robots path.

O. Castillo et al. (Eds.): Recent Advances on Hybrid Intelligent Systems, SCI 451, pp. 21–31.
springerlink.com

The reactive controller is for those cases where an obstacle free path cannot be guaranteed; this is where the reactive controller will do its work providing a behavior that will make the robot react to any type of obstacle so that the robot can continue on its journey. In this paper we describe the integration method for these two controls as part of the complete Navigation Control System, the control blocks are fuzzy inference systems of type-1 and type-2, and a general GA (Genetic Algorithm) is applied to the optimization of each of the controller blocks with a specific fitness function for each part that will evaluate the corresponding individual performance.

As related work, we can find that of Cupertino et al [8] developed a Fuzzy controller of a mobile robot, based on 3 FLCs (Fuzzy Logic Controller) and one Fuzzy Supervisor that was in charge of determining which FLC behavior will be active, there the FLCs are of Type-1 and the Fuzzy Supervisor mainly acts as a switch. In our proposed method the fuzzy integrator acts more like a fusion block. S. Coupland et al [7] proposed a Type-2 Fuzzy Control of a Mobile Robot, which is based on W Payton et al [22] Command Fusion, where the idea is that a behavior should work with others to find a mutually beneficent solutions, where each behavior takes into consideration every possible output with its corresponding activation value (positive or negative), and a winner takes all network is use to select the winning responses for each behavior. Coupland suggests using two FISs one for goal seeking and another for obstacle avoidance. The activation value for each of the FIS output will be a Fuzzy set that will be passed to the command fusion block to later be defuzzifed and that crisp value pass to the Actuator block, being a difference with our proposed control the integration method of the two behaviors. The control navigation of a mobile robot is a topic that has been extensively investigated over the years, the method proposed in this paper is based on the idea that separation and the cooperation between key behaviors produces a better result than the use of a single behavior and it differs from previous approaches from the integration perspective done by a FIS that is in charge of the weighted system, that will assign a weight to each response from each controller by each control step that is combined to obtain a unified single response to the robot.

This paper is organized as follows: In section 2 we describe the mobile robot used in these experiments, section 3 describes the development of the evolutionary method. Section 4 shows the simulation results. Finally, section 5 shows the Conclusions.

2 Mobile Robot

The particular mobile robot considered in this work. The robot is based on the description of the Simulation toolbox for mobile robots [26], which assumes a wheeled mobile robot consisting of one conventional, steered, unactuated and not-sensed wheel, and two conventional, actuated, and sensed wheels (conventional wheel chair model). This type of chassis provides two DOF (degrees of freedom) locomotion by two actuated conventional non-steered wheels and one unactuated steered wheels. The Robot has two degrees of freedom (DOFs): y-translation and either x-translation or z-rotation [26]. Fig. 1 shows the robot's configuration, it has 2 independent motors located on each side of the robot and one castor wheel for support located at the front of the robot.

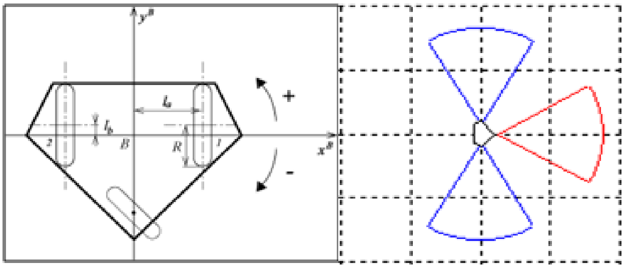


Fig. 1. Kinematic coordinate system assignments[26]

The kinematic equations of the mobile robot are as follows:

Eq. 1: The sensed forward velocity solution [26]

$$\begin{pmatrix} V_{B_x} \\ V_{B_y} \\ \omega_{B_z} \end{pmatrix} = \frac{R}{2l_a} \begin{bmatrix} -l_b & l_b \\ -l_a & -l_a \\ -1 & -1 \end{bmatrix} \begin{pmatrix} \omega_{W_1} \\ \omega_{W_2} \end{pmatrix} \tag{1}$$

Eq. 2: The Actuated Inverse Velocity Solution [26]

$$\begin{pmatrix} \omega_{W_1} \\ \omega_{W_2} \end{pmatrix} = \frac{1}{R(l_b^2+1)} \begin{bmatrix} -l_a l_b & -l_b^2 - 1 & -l_a \\ l_a l_b & -l_b^2 - 1 & l_a \end{bmatrix} \begin{pmatrix} V_{B_x} \\ V_{B_y} \\ \omega_{B_z} \end{pmatrix} \tag{2}$$

Under the Metric system are define as:

V_{B_x}, V_{B_y} Translational velocities $[\frac{m}{s}]$,
ω_{B_z} Robot z-rotational velocity $[\frac{rad}{s}]$,
$\omega_{W_1}, \omega_{W_1}$ Wheel rotational velocities $[\frac{rad}{s}]$,
R Actuated wheel radius[m],
l_a , l_b Distances of wheels from robot's axes [m].

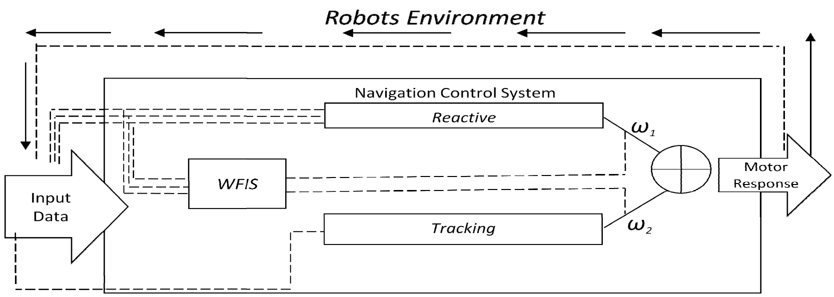
3 Navigation Control System

The proposed control system consists of three main fuzzy blocks, two are behavior based and the other one is in charge of the response integration, the behaviors are the reactive and tracking blocks, and each one will provide its specific behavior that will be combined into one response by the integration block.

Each behavior block is in charge of its own task, the problem is that they seem to be in conflict with each other when an unexpected obstacle arises, because if at the time of planning the route the obstacles are present then the route can be designed to avoid them, but when there are obstacles that we where un aware off, the two behaviors enter in contradiction one is designed to avoid the object and the other to keep the robot on its track.

The most common solution will be to just switch between controllers when need it, however, this approach is not very efficient due to the lack of awareness the two blocks have of each other, the reactive will effectively keep the robot from the collision but it may redirect the robot farther away from its destination to a point where the tracking controller can no longer find its way back to the reference, or the tracking controller can guide the robot straight into the obstacle if the reactive control is not activated on time. The proposed referral for control navigation is to always have both controls active and their responses are combined and generate the movement of the robot, the integration is done with another fuzzy block call WFIS[15] (Weight-Fuzzy Inference System) and what this controller does is to assign response weights to each of the controllers crisp response value.

The inputs are gathered from the information that we can collect from the robot (sensors) or the environment by other means (cameras) and from this we need to create the knowledge rule base to give higher activation values to the response we want to take the lead on the robot movement one example of the rule is the following (*if Front_Sensor_Distance is Close Then TranckingWeight is Medium and ReactiveWeight is Medium*), both off our controls provide the right and left motor speed and we combine each one with the weight given by the WFIS block. Figure 2 shows the proposed navigation control.



$$RMSR = \frac{\left[\left(\frac{\sum_i \mu_A(y_i)\times y_i}{\sum_i \mu_A(y_i)}\right)*\omega_1\right] + \left[\left(\frac{\sum_i \mu_A(y'_i)\times y'_i}{\sum_i \mu_A(y'_i)}\right)*\omega_2\right]}{2}$$

$$LMSR = \frac{\left[\left(\frac{\sum_i \mu_A(y'''_i)\times y'''_i}{\sum_i \mu_A(y'''_i)}\right)*\omega_1\right] + \left[\left(\frac{\sum_i \mu_A(y''_i)\times y''_i}{\sum_i \mu_A(y''_i)}\right)*\omega_2\right]}{2}$$

Fig. 2. Navigation Control System [15]

4 Genetic Algorithms

The Genetic Algorithm (GA) was applied to each of the design problems, of finding the best fuzzy reactive and tracking controllers [16]; however this paper will only focus on the WFIS controller.

The purpose of using an evolutionary method is to find the best possible controllers of each type and this can be obtained using the GA, as it searches along the solution space, combining the attributes from the best controllers in generating new ones, this concept taken from the building blocks theory.

The idea was to optimize the parameters in the Membership Functions, but also the number of Membership functions and this means to also optimize the number of rules making this a multi objective problem. For this we will take advantage of the HGA (Hierarchical Genetic Algorithm) intrinsic characteristic to solve multi objective problems.

The work of the GA was divided in two main modules, one that handles all the operations related to the selection and chromosome manipulation, which we use for all our controllers that we work on, the other module is the one where we evaluated the performance of each chromosome and this part is different on each case. With this approach we utilize the generality of the GA and just have a specific evaluation method for each controller. Figure 3 shows the 2 main modules.

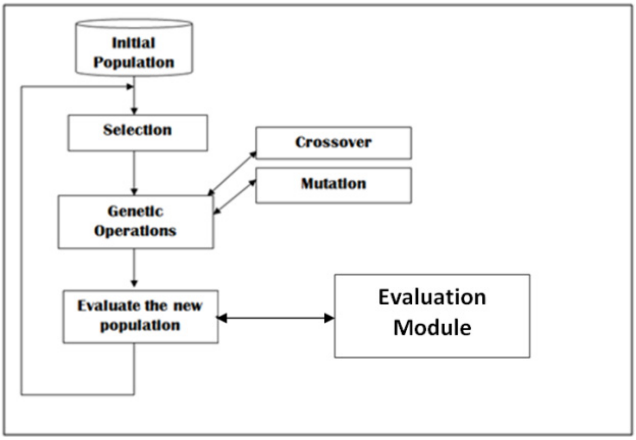


Fig. 3. Genetic Algorithm process

The GA module is in charge of initializing the population, selecting the chromosomes that will be used for the genetic operations and letting the Evaluation Module know which chromosomes are ready to be evaluated and reinserting them to the population pool.

4.1 Chromosome Encoding

Each individual on the population will represent a FIS controller, each of which will be encoded on a vectorial structure that will have "n" main sections, one for each variable (input and output). Each main section will contain 2 subsections (control genes, Connection genes). The section and subsection sizes depend on the controller that they represent.

4.2 WFIS Controller

The function of the WFIS control is to correctly combine the 2 behaviors of tracking and reaction and obtain a new global behavior that resembles the same ability that we apply when we are driving, that is to react to unexpected objects, but in a more basic concept and ability, to the problem that is the navigation of the robot. A forward moving behavior response out off the global control is desired. The objective is to guide the robot through the reference avoiding any collision with any obstacle present. It's not our objective to optimize the robot to find the maze exit, we use a closed space where the robot cannot easily wonder off and each wall is considered an obstacle to the robot that it must avoid while it moves around. The FISs are Mamdani type-1 fuzzy systems [14], each consisting of 3 inputs, which are the distances obtained by the robots sensors described on section 2, and 2 outputs that are the weights that will be used to integrate the responses of the other 2 controllers, all this information is encoded into each chromosome.

4.2.1 Type-1 Fuzzy Weight Controller Chromosome Architecture

The control genes consist of 5 bit vectors, this will indicate which fuzzy membership is or not active, the connection genes are divided in 5 subsections, 5 is the maximum number of membership functions that are allowed per variable, each of which can be trapezoidal or triangular membership function, and each of these subsection is divided into 2 sections one that indicates the type of the membership function and the other the parameters for the function, see Figure 4.

4.3 WFIS Controller Objective Function

The WFIS controller performance is measured with the RMSE between the reference and the robots trajectory. We apply the test three times and take the average, on each of the three tests the robot and the reference vertical position is random, but we ensure that on one test the robots vertical position is above the reference and on another test is below it, we do this to ensure the controller works properly for any case the robot may need it when it´s above or below (Figure 5).

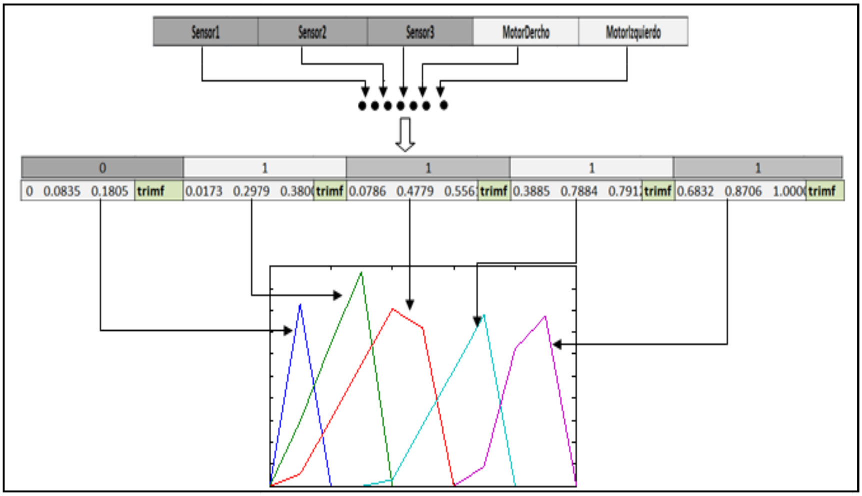


Fig. 4. Type-1 WFIS Controller Chromosome Architecture

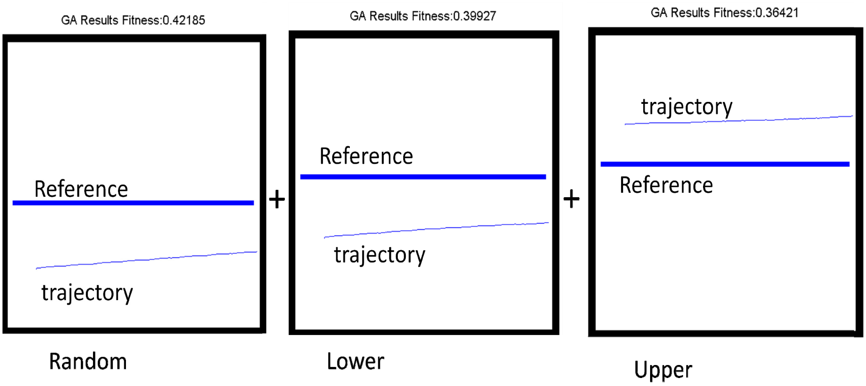


Fig. 5. Fitness Functions for the WFIS Controller

5 Simulation Results

For the simulation experiment the GA and the evaluation process were separated into two different parts, the generic GA process was developed on the C# language with .net 4, where a GA and Fuzzy System library where created with a GUI to setup the GA parameters, there the GA operations and cycle are run and the FIS are created. When a chromosome is ready to be evaluated it lets Matlab know and a modified version of the Simulation toolbox for mobile robots [26] is used to run each test, where the performance is measured and a Fitness value is returned to the GA process, and the communication between both process is done using a SQL server queue table.

5.1 WFIS Controller

For the type-1 WFIS controller, a GA was setup with high number of generations and a low value of population size, this because of the large solution space. The reasoning behind this is that with a relative small group of individuals it will cover focused sections of the solution and can move around the space, A constrain for inputs and outputs of maximum 10 and minimum 2 FMS was set, on the outputs, the evaluation as described on section 4 is based upon each individual performance on the particular maze problem.

Table 1. Summary of Type-1 WFIS Results

	Membership Chromosome		Fuzzy Rule Chromosome
	Control Genes	Connections Genes	
Representation	Binary	Real Number	Integer
Population Size	10		
No. of Offspring	3		
Crossover	One Point	One Point	
Crossover Rate	1.0	1.0	
Mutation	Bit Mutation	Random Mutation	Shift index operation
Mutation Rate	0.02	0.02	
GA Parameters			
Generation	1500		
Selection	Roulette Wheel with Ranking		
Results			
Rank	Fitness	Active FM's ($S_1+S_2+S_3+W_1+W_2$)	Active Rules
1	0.2393	(3+4+3+3+3)=16	36
2	0.2450	(3+4+3+3+3)=16	36
3	0.2514	(4+5+4+4+3)=20	80
4	0.2543	(4+3+3+4+3)=17	36
5	0.2551	(3+4+3+4+3)=17	36
6	0.2633	(4+4+3+4+4)=19	48
7	0.2643	(4+3+4+3+5)=19	48
8	0.2691	(3+3+3+4+3)=16	27
9	0.2742	(3+3+3+4+3)=16	27

Table 1 shows the GA configuration and the top 9 Results, where we have the fitness value and the number of membership functions of each input and output, where the S represents the inputs and indicate the sensor number and W the outputs and indicate the Weight number, and the total rules that are active on each controller.

Fig 6 shows the set of 3 tests during the evaluation process of the GA, where the red line is the reference, the blue dotted line is robot path on each run and the gray squares are obstacle located around the reference path.

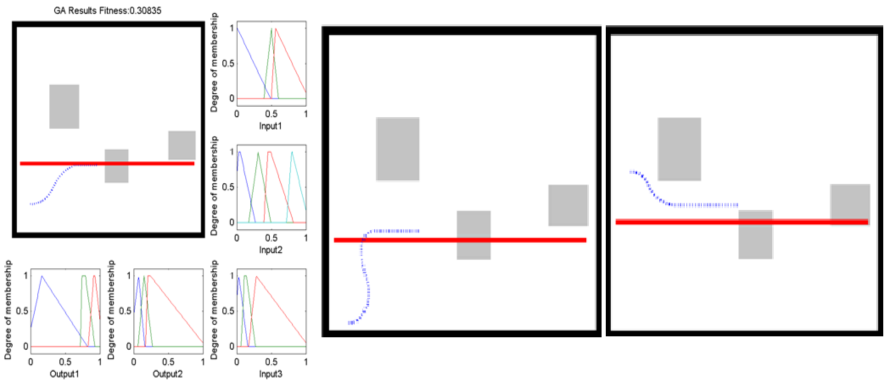


Fig. 6. Type-1 WFIS Controller Results

6 Conclusions

In this paper we have been able to optimize the Type-1 and Type-2 Reactive and Tracking Controllers and developed a GUI to optimize the Fuzzy Inference System, and we are currently working on the Type-1 WFIS Optimization.

Future work will consist in the Optimization of the WFIS based on a Type-2 fuzzy system.

Acknowledgment. We would like to express our gratitude to CONACYT, and Tijuana Institute of Technology for the facilities and resources granted for the development of this research.

References

[1] Aceves, A., Aguilar, J.: A Simplified Version of Mamdani's Fuzzy Controller The Natural Logic Controller. IEEE Transactions on Fuzzy Systems 14(1), 16–30 (2006)
[2] Aguilar, L., Melin, P., Castillo, O.: Intelligent Control of a stepping motor drive using a hybrid neuro-fuzzy ANFIS approach. Applied Soft Computing 3(3), 209–219 (2003)
[3] Astudillo, L., Castillo, O.T., Aguilar, L.: Intelligent Control of an Autonomous Mobile Robot Using Type-2 Fuzzy Logic. Engineering Letters 13(2), 93–97 (2006)
[4] Bell, M., Toriu, T., Nakajima, S.: Image-Based Robot Map Building and Path Planning with an Omnidirectional Camera Using Self-Organising Maps. International Journal of Innovative Computing, Information and Control, 3845–3852 (2011)

[5] Castillo, O., Melin, P.: New fuzzy-fractal-genetic method for automated mathematical Modeling and Simulation of Robotic Dynamic Systems. IEEE, International Conference on Fuzzy Systems, vol. 2, 1182–118 (1998)
[6] Castillo, O., Melin, P.: New fuzzy-fractal-genetic method for automated mathematical Modeling and Simulation of Robotic Dynamic Systems. In: IEEE, International Conference on Fuzzy Systems, vol. 2, pp. 1182–1118 (1998)
[7] Coupland, S.: Type-2 Fuzzy Control of a Mobile Robot. PhD Transfer Report. De Montfort University, UK (2003)
[8] Cupertino, F., Giordano, V., Naso, D., Delfine, L.: Fuzzy control of a mobile robot. IEEE Robotics & Automation Magazine, 74–81 (2006)
[9] Fate, M.: Robust Voltage Control of Electrical Manipulators in Task-Space. International Journal of Innovative Computing, Information and Control, 2691–2700 (2010)
[10] Ishikawa, S.: A Method of Indoor Mobile Robot Navigation by Fuzzy Control. In: Proc. Int. Conf. Intell. Robot. Syst., Osaka, Japan, pp. 1013–1018 (1991)
[11] Klir, J.G., Yuan, B.: Fuzzy Sets, Fuzzy Logic, and Fuzzy Systems: Selected Papers by Lotfi A. Zadeh. In: Advances in Fuzzy Systems: Application and Theory, vol. 6. World Scientific Publishing Company (1996)
[12] Kim, C., Lee, K.: Robust Control of Robot Manipulators Using Dynamic Compensators under Parametric Uncertainty. International Journal of Innovative Computing, Information and Control, 4129–4137 (2011)
[13] Leyden, M., Toal, D., Flanagan, C.: A Fuzzy Logic Based Navigation System for a Mobile Robot. In: Proceedings of Automatisierungs Symposium (1999)
[14] Mamdani, E.: Applications of fuzzy logic to approximate reasoning using linguistic synthesis. In: Proc. 6th Int. Symp. on Multiple Value Logic, pp. 196–202. Utah State University (1976)
[15] Melendez, A., Castillo, O., Soria, J.: Reactive and Tracking Control of a Mobile Robot in a Distributed Environment Using Fuzzy Logic. In: IEEE International Conference on FUZZ, pp. 1–5 (2010)
[16] Melendez, A., Castillo, O., Melin, P.: Evolutionary Optimization of the Fuzzy Controllers in a Navigation System for a Mobile Robot. International Journal of Innovative Computing, Information and Control, International Journal (in press)
[17] Meléndez, A., Castillo, O., Soria, J.: Reactive Control of a Mobile Robot in a Distributed Environment Using Fuzzy Logic. In: Fuzzy Information Processing Society, NAFIPS 2008. Annual Meeting of the North American, vol. (19-22), pp. 1–5 (2008)
[18] Melin, P., Castillo, O.: Intelligent Systems with Interval Type-2 Fuzzy Logic. International Journal of Innovative Computing, Information and Control, 771–784 (2008)
[19] Melin, P., Castillo, O.: Intelligent control of aircraft dynamic systems with a new hybrid neuro-fuzzy fractal approach. Information Sciences 142(1-4), 161–175 (2002)
[20] Melin, P., Castillo, O.: Adaptive Intelligent control of aircraft systems with a hybrid approach combining neural networks, fuzzy logic and fractal theory. Applied Soft Computing 3(4), 353–262 (2003)
[21] Pishkenari, H.N., Mahboobi, S.H., Meghdari, A.: On the Optimum Design of Fuzzy Logic Controller for Trajectory Tracking Using Evolutionary Algorithms. In: Cybernetics and Intelligent Systems, 2004 IEEE Conference on Publication Date: 1-3, vol. 1, pp. 660–665 (2004)
[22] Payton, D.W., Rosenblatt, J.K., Keirsey, D.M.: Plan guided reaction. IEEE Transactions on Systems, Man and Cybernetics 20(6), 1370–1382 (1990)

[23] Shafiei, S., Soltanpour, M.: Neural Network Sliding-Mode-PID Controller Design for Electrically Driven Robot Manipulators. International Journal of Innovative Computing, Information and Control, 511–524 (2011)
[24] Shafiei, S., Soltanpour, M.: Robust Task-Space Control of Robot Manipulators under Imperfect Transformation of Control Space. International Journal of Innovative Computing, Information and Control, 3949–3960 (2009)
[25] Thomson, A., Baltes, J.: A path following system for autonomous robots with minimal computing power. University of Auckland, Private Bag 92019, Auckland, New Zealand, Technical Report (2001)
[26] Mobile robotics toolbox for Matlab 5 (2001), `http://www.uamt.feec.vutbr.cz/robotics/simulations/amrt/simrobot_en.html`