

An Optimized Interval Type-2 Fuzzy Logic Control Scheme based on Optimal Defuzzification

Ziyad T. Allawi
College of Education for
Humanitarian Studies,
University of Baghdad, Iraq

Turki Y. Abdalla
Department of Computer Engineering,
College of Engineering,
University of Basrah, Iraq

ABSTRACT

In this paper, a new way of optimizing fuzzy logic is introduced. This way is used to optimize the output of Interval Type-2 Fuzzy Logic controller by replacing the Defuzzification stage by the Optimization algorithm. The algorithm chooses the best crisp output variable from the type-reduced set which is the output of the Type-Reduction stage instead of averaging the set extremes which was performed by Defuzzification stage. Artificial Bee Colony optimization algorithm is used to optimize the Interval Type-2 Fuzzy Logic controller to manage the navigation of multiple mobile robots in indoor environments.

Keywords

Bio-inspired Optimization; Type-2 Fuzzy Logic; Differential Drive Mobile Robots; Navigation

1. INTRODUCTION

The Type-2 Fuzzy Logic Control System (T2 FLS) is a modern class and an extension of the well known fuzzy logic control system. The first concepts about T2 FLS were introduced by Professor Zadeh in [1]. The original fuzzy logic (Type-1 FLS) is unable to handle uncertainties because the word “fuzzy” has the notation of uncertainty. The extended fuzzy logic (T2 FLS) is able to handle uncertainties because it can model and minimize their effects.

The first concrete research papers about T2 FLS were introduced by Karnik and Mendel [2], [3] and Karnik et al. [4]. In these papers, the fundamentals of T2 FLS were introduced, like development of “Type-Reduction”; set operations; properties of membership grades; relations, compositions and defuzzification. Type-Reduction methods are computationally intensive, so the interval Type-2 fuzzy logic was introduced because it provides a simple computation procedure.

Liang and Mendel [5] introduced the theory and design procedures of Interval Type-2 Fuzzy Logic Systems (IT2 FLS). The IT2 FLS has always been considered to be a special case of a general T2 FLS; consequently, things that were developed for the latter were then specialized to the former. Research works about developing IT2 FLS were presented since Mendel et al. [6] have proposed a simple way to implement an IT2 FLS from T1 FLS mathematics. The use of IT2 FLS began to increase since then. Some papers [7] – [9] investigated the importance of IT2 FLS theoretically and practically, especially in intelligent control systems.

Swarm Intelligence (SI) deals with the collective behavior of swarms through complex interaction of individuals without supervision. Under the term of SI lie the Bio-inspired

Optimization Algorithms which are population-based meta-heuristic optimization algorithms. They include bee or ant colonies, bird flocking, animal herding, bacterial growth, and fish schooling. Some of these algorithms are the Ant Colony Optimization (ACO), Particle Swarm Optimization (PSO) and Artificial Bee Colony (ABC) algorithms. Most of these methods have been developed only in recent years and are emerging as popular methods for the solution of complex engineering problems [10].

ABC algorithm is one of the modern global optimization algorithms. It is as an optimization tool which provides a population-based search procedure in which individuals called food positions are modified by the artificial bees with time and the bee’s aim is to discover the places of food sources with high nectar amount and finally the one with the highest nectar. ABC uses only common control parameters such as colony size and maximum cycle number.

ABC was introduced by Karaboğa [11] in 2005. He presented it as a branch of the SI. He then continued the development of ABC algorithm and published three papers with Başturk [12] – [14]. They compared ABC performance with GA and PSO on a group of benchmark numerical multi-dimensional functions. They found that ABC outperformed all other mentioned algorithms.

Several modifications and improvements were carried out upon ABC. Karaboğa and Akay [15] modified ABC to be executable for constrained optimization problems. In [16], Karaboğa; again, modified ABC by introducing a new parameter to control the frequency of perturbations on the predicted solutions.

In this paper, ABC will be used to optimize the output signal of the Type-2 fuzzy controller. ABC will use the output of the Type-reduction stage as a solution space and then will select a solution which minimizes some objective function. This algorithm will replace the Defuzzification stage which generates the output from the average of the solution interval. This controller will be used to control the navigation of multiple mobile robots in an indoor environment.

2. RELATED WORKS

In the field of Mobile Robotics, T2 FLS began to take over T1 FLS because the former can handle system uncertainties better than the latter. In the literature, Hagrass [17] presented a novel reactive control architecture for autonomous mobile robots that was based on T2 FLS to implement the basic navigation behaviors and the coordination between these behaviors. Astudillo et al. [18] developed a tracking T2 FLS controller for the dynamic model of unicycle mobile robot by integrating a kinematic controller and a torque controller based on Fuzzy

Logic Theory. Martinez et al. [19], [20] made the same work as Astudillo but with adding GA for optimization. Leottau and Melgarejo [21] presented an approach for designing an IT2 FLS for a mobile robot application and described how it could be developed involving the use of T1 and T2 fuzzy sets. Junratanasiri et al. [22] proposed a navigation system in an uncertain environment focusing on dynamic obstacles for a mobile robot. IT2 FLS was utilized to compute the linear and angular velocities of a mobile robot.

Although the optimization of fuzzy systems has begun with GA since 1991 [23]; the concern will be here on optimization of fuzzy logic systems using bio-inspired optimization algorithms. Shi and Eberhart [24] were the first who embed the fuzzy logic in the optimization algorithm. They used the fuzzy logic to dynamically adapt the inertia weight of the fundamental function of PSO algorithm. Tian and Li [25] improved the work in [24] by fuzzifying the learning rate as well as the inertia weight.

Wong et al. [26] integrated the fuzzy systems with PSO for mobile robot navigation. They proposed a PSO-based method to automatically determine appropriate membership functions (MF's) of the fuzzy controller to determine velocities of the left and right-wheeled motors of the two-wheeled mobile robot, so that the controlled robot can move to any desired position effectively in a two-dimensional space.

Oscar Castillo led the way in the field of T1 and T2 FLS optimization by using bio-inspired algorithms. In his paper with Valdez and Melin [27], they described a new hybrid approach for mathematical function optimization combining PSO and GA using ordinary fuzzy logic to combine the results of the PSO and GA in the best way possible. Castillo, with Martinez et al. [28], applied evolutionary methods for the optimization of T2 FLS. These optimal controllers were used for the trajectory tracking control of autonomous mobile robots. Another paper proposed by Castillo et al. [29] describes the application of GA, ACO and PSO on the optimization of the MF's parameters of T1 and T2 FLS in order to find the optimal intelligent controller for an autonomous wheeled mobile robot. Results indicated that ACO outperforms PSO and GA, respectively.

Integration of the ABC algorithm with fuzzy-based systems and mobile robotics is still in the early stages. Few research presents integrating ABC for optimizing the rule base in fuzzy controllers. One of these researches was O. Abedinia et al. [30]. Bhattacharjee et al. [31] used ABC for optimizing multi-robot path planning. The proposed method tries to determine a trajectory of motion to known targets where ABC is used to minimize the path through obstacle and collision avoidance.

3. RESEARCH METHOD

3.1 Type-2 Fuzzy Logic Systems

A FLS described using at least one Type-2 fuzzy set is called a Type-2 FLS (T2 FLS). It is characterized by fuzzy membership grades. An interval Type-2 fuzzy set \tilde{A} in the domain X is defined as [32]:

$$\tilde{A} = \int_{x \in X} \int_{u \in J_x \subseteq [0,1]} 1/(x, u) = \int_{x \in X} \left[\int_{u \in J_x \subseteq [0,1]} 1/u \right] / x \quad \dots(1)$$

In (1), x is the primary variable belongs to X ; u is the secondary variable, which has domain J_x at each $x \in X$; J_x is called the primary membership of x . For interval Type-2 sets, the secondary grades of \tilde{A} all equal 1.

Reference [32] contains more information about T2 FLS.

3.2 Interval Type-2 Fuzzy Inference System:

An interval Type-2 FLS (IT2 FLS), which is a FIS that uses at least one interval Type-2 fuzzy set, contains five components: fuzzifier, rule base, inference system, type-reducer, and defuzzifier.

Fig. 1 shows a block diagram of an IT2 FLS.

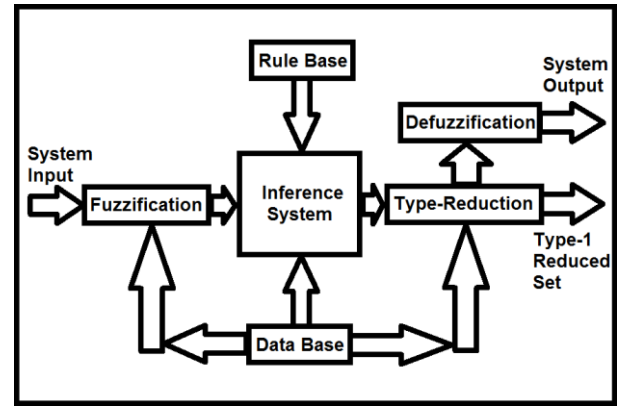


Fig 1: Interval Type-2 Fuzzy Logic System

The outputs of the IT2 FLS inference system are Type-2 fuzzy sets. These sets must be type-reduced before the defuzzifier can be used to generate a crisp output. This is the main structural difference between Type-1 and Type-2 FLS's. The most commonly used Type-Reduction method is the Center of Sets (COS) type-reducer, which was expressed in [3]. The output of the Type-reduction stage is an interval Type-1 set $Y = [y_l, y_r]$ corresponding to the centroid of the interval Type-2 consequent set. This method could be computed by using the slow Karnik-Mendel algorithm (KMA) or by using the fast Enhanced Iterative Algorithm with Stop Condition (EIASC) which was introduced by Wu and Nie [33]. This algorithm saves time more than KMA in about 70% especially when the number of MF's is less than 100.

Once y_l and y_r are obtained, they can be used to calculate the crisp output. Since the Type-reduced set is an interval Type-1 set, the defuzzified output is:

$$y(x) = \frac{y_r(x) + y_l(x)}{2} \quad \dots(2)$$

3.3 Artificial Bee Colony:

ABC algorithm is one of the current global optimization algorithms. It is as simple as ACO and PSO, and uses some general control parameters such as colony size and maximum cycle number. ABC as an optimization tool provides a population-based search process in which individuals called foods positions are modified by the artificial bees with time and the bee's aim is to find out the places of food sources with high nectar quantity and finally the one with the highest nectar. In ABC system, artificial bees fly around in a multidimensional search space and some (employed and onlooker bees) choose food sources depending on the knowledge of themselves and their nest mates, and regulate their positions. Some (scouts) fly and select the food sources randomly without using experience. If the nectar amount of a new source is higher than that of the previous one in their

memory, they remember the new position and forget the previous one.

ABC system thus combines local search methods, carried out by employed and onlooker bees, with global search methods, managed by onlookers and scouts, attempting to balance exploration and exploitation process [14].

ABC algorithm procedure is explained in details in [15].

3.4 Optimization of Type-2 Fuzzy Logic System:

The ABC optimization algorithm will be used in this article to optimize IT2 FLS controller which is used for control the navigation of Multiple Mobile Robot Systems (MMRS). In this situation, the defuzzification stage is replaced by the optimization algorithm as in Fig. 2.

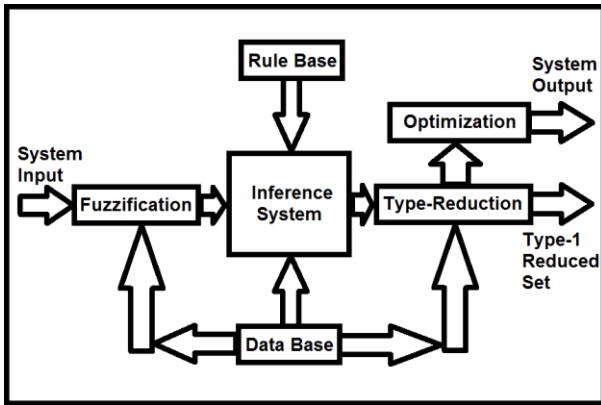


Fig 2: Optimization of Interval Type-2 Fuzzy Logic System

The reason behind replacing defuzzification stage by the optimization algorithm is that the output of the Type-Reduction process is a Type-1 reduced output set. This output set is an interval-type output and any output value located in that interval set can be chosen as a crisp output. The rule of the optimization algorithm is essential here to help in choosing the optimum output value based on an objective function. The objective function used in this study is as below:

$$fitness_i = \frac{1}{2} \sum_{j=1, j \neq i}^n \left(\frac{1}{tc_{ij}} \right)^2 + |\angle \mathbf{v}_{target} - \angle \mathbf{v}_i| \quad \dots(3)$$

where n is the number of the robots in the environment, i and j are robot indices between 1 and n . tc_{ij} is the time-to-collision between robot i and robot j which can be computed through geometrical manner. The second term of the fitness function refers to the closeness of velocity vector to the target direction. The resultant fitness of the robot i is the sum of all of its mutual fitness values with other robots independently.

4. DESIGN PROCEDURE

The mobile robot used in the design procedure is the E-puck mobile robot designed at the labs of the *Swiss Federal Institute of Technology in Lausanne* (EPFL) [34]. The controller which is used to control the E-puck is an ABC optimized IT2 FLS controller. The IT2 FLS accepts the inputs from the E-puck's two front sensors "ps0 and ps7", from the right sensor "ps1" and from the left sensor "ps6"; then the MF's values of these sensors are valued. If the sensor reading

is below 100, the obstacle is regarded as 'far'; however, if the sensor reading is above 500, the obstacle is regarded as 'near'. Fig. 3 illustrates this.

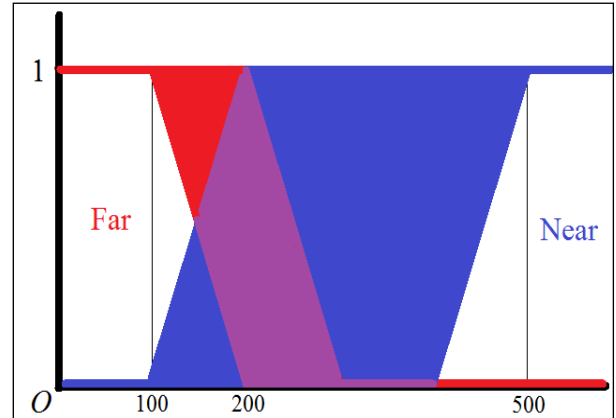


Fig 3: MF's of the Sensor Data IT2 FLS Input Variables

In the E-puck fuzzy rule-base, 16 rules are needed to fulfill all the controller possibilities; because there are four input variables with 2 MF's for each variable. When the two front proximity sensors detect an obstacle and their input values are 'Near', the robot turns around towards its open side, either to the left or to the right. However, if one of the front sensors is 'Near' and the other is 'Far', the robot turns around to the side which has the 'Far' value. If both of the two front sensors are 'Far', it checks the other two side sensors. If one of them is 'Near' and the other is 'Far', the robot turns around towards the open side, either to the left or to the right. If all of the sensors read 'Far', the robot then moves forward.

According to the previous paragraph, the rule-base table of the E-puck DDMR could be designed. The complete Rule-Base Table of the IT2 FLS controller which is used in the Webots™ simulation software [35] is shown in Table 1.

Table 1: Rule Table of IT2 FLS Controller for the E-puck Mobile Robot

Rule No.	Antecedents				Consequents	
	ps7	ps0	ps6	ps1	LW	RW
1	Near	Near	Near	Near	B	F
2	Near	Near	Near	Far	F	B
3	Near	Near	Far	Near	B	F
4	Near	Near	Far	Far	B	F
5	Near	Far	Near	Near	F	B
6	Near	Far	Near	Far	F	B
7	Near	Far	Far	Near	F	B
8	Near	Far	Far	Far	F	B
9	Far	Near	Near	Near	B	F
10	Far	Near	Near	Far	B	F
11	Far	Near	Far	Near	B	F
12	Far	Near	Far	Far	B	F
13	Far	Far	Near	Near	B	F
14	Far	Far	Near	Far	F	B
15	Far	Far	Far	Near	B	F
16	Far	Far	Far	Far	F	F

The IT2 FLS generates two outputs, which are the *Left Wheel Speed* (LW) and the *Right Wheel Speed* (RW). Each output has two singleton MF's which are 'Forward' (F) and 'Backward' (B). Fig. 4 illustrates the output MF's.

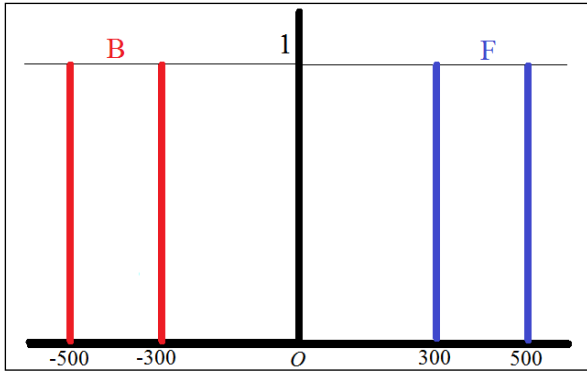


Fig 4: MF's of the Motor Speed IT2 FLS Output Variables

The target direction is obtained from the camera or from another robot through Radio link. The output speed is chosen according to the target direction by the optimization algorithm.

5. RESULTS AND DISCUSSIONS:

In the simulation scenario of ABC-optimized IT2 FLS algorithm using Webots™, 8 E-puck robots are used. In this scenario, the robots do not know where their targets are; but they know some of their features. The target color for the robots is green and located in the middle of the environment. The robots use their cameras to search for their targets. They use Radio links to communicate with each other and share information about the possible targets. When a robot finds a target, it modifies its heading; and then heads directly towards the target until it becomes close enough to it. The robot then stops moving and inform the other robot about its location.

The parameters of the ABC algorithm which were used in the simulation are: Population Size: 100 bees, Maximum Iterations: 250, and the stopping condition is due when all bees converge to one solution with tolerance of $1e-3$ OR maximum iterations reached [12].

The robots start moving in the environment as shown in Fig. 5, when one of the robots found the target; it informs the others about its location. So, all the robots start to move towards the found target until they reach to it as shown in Fig. 6.

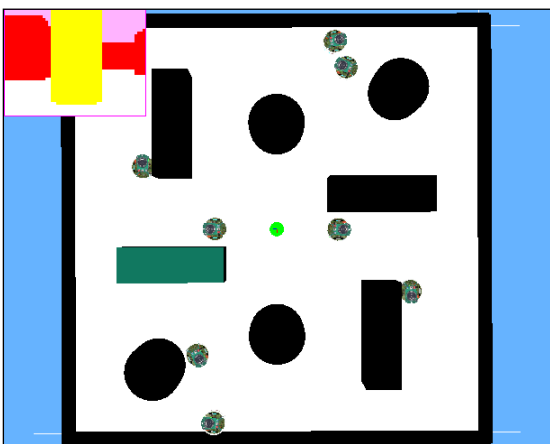


Fig 5: One of the robots found the target and informed the others about its location.

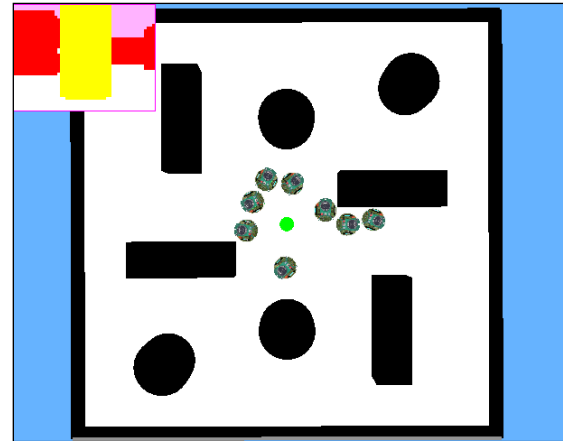


Fig 6: All the robots reached the target

For comparison, the same scenario was performed using the ordinary defuzzification method in the IT2 FFLS controller. The results of the travelled distance and robot speed for the two controllers are shown in Table 2.

Table 2: Comparison between the Ordinary and Optimized IT2 FLS controller

Robot No.	Ordinary Controller		Optimized Controller	
	Distance	Speed	Distance	Speed
R1	1.38	3.9	1.28	5.5
R2	1.24	4.2	1.04	5.1
R3	1.36	4.1	1.21	5.2
R4	1.02	3.8	0.98	5.0
R5	0.78	4.1	0.72	5.3
R6	0.82	4.0	0.81	5.2
R7	0.33	5.0	0.33	6.3
R8	0.33	5.0	0.33	6.3

Table 2 shows that the robots have reached the target in less distance and faster speed for all of them. This means that the optimized controller has succeeded to select the best velocity from the type-reduced set; as compared with the ordinary controller which selects the average of the interval extremes.

6. CONCLUSIONS

It can be concluded that this optimization technique has improved the output signal of the fuzzy controller. The velocity output of the optimized controller produced better results than the ordinary controller. Also, the optimization was performed only upon the output parameter whereas the optimization of the intermediate parameters was not needed. This makes the optimization process faster and suitable for online processes.

Future suggests improving this method by using AI strategies like Petri nets rather than optimization algorithms for fast determination of optimum robot speed parameters.

7. REFERENCES

- [1] Zadeh, L. A., "The Concept of a Linguistic Variable and Its Application to Approximate Reasoning," IEEE Journal of Information Science, Vol. 8, 1975, pp. 199-249.
- [2] Karnik, N. N. and Mendel, J. M., "Introduction to Type-2 Fuzzy Logic Systems," IEEE Fuzzy Conference, Anchorage, USA, 1998, pp. 915-920.

- [3] Karnik, N. N. and Mendel, J. M., "Type-2 Fuzzy Logic Systems: Type Reduction," IEEE Journal of Systems, Man and Cybernetics, Vol. 2, 1998, pp. 2046-2051.
- [4] Karnik, N. N. and et al., "Type-2 Fuzzy Logic Systems," IEEE Transactions on Fuzzy Systems, Vol. 7, No. 6, 1999, pp. 643-658.
- [5] Liang, Q. and Mendel, J. M., "Interval Type-2 Fuzzy Logic Systems: Theory and Design," IEEE Transactions on Fuzzy Systems, Vol. 8, No. 5, 2000, pp. 535-550.
- [6] Mendel, J. M. and et al., "Interval Type-2 Fuzzy Logic Systems Made Simple," IEEE Transactions on Fuzzy Systems, Vol. 14, No. 6, 2006, pp. 808-821.
- [7] Mendel, J. M., "On the Importance of Interval Sets in Type-2 Fuzzy Logic Systems," IFSA World Congress and 20th NAFIPS International Conference, Vancouver, Canada, 2001, pp. 1647-1652.
- [8] Castro, J. R. and Castillo, O., "Interval Type-2 Fuzzy Logic for Intelligent Control Applications," IEEE Annual Meeting of the North American Fuzzy Information Processing Society, San Diego, California, USA, 2007, pp. 592-597.
- [9] Castillo, O., "Interval Type-2 Fuzzy Logic for Control Applications," IEEE International Conference on Granular Computing, San Jose, California, USA, 2010, pp. 79-84.
- [10] Rao, S. S., Engineering Optimization, John Wiley & Sons Inc., Hoboken, New Jersey, USA, 2009.
- [11] Karaboğa, D. "An Idea Based On Honey Bee Swarm for Numerical Optimization," Technical Report-TR06, Computer Engineering Department, Engineering Faculty, Erciyes University, Kayseri, Turkey, 2005.
- [12] Karaboğa, D. and Başturk, B., "A Powerful and Efficient Algorithm for Numerical Function Optimization: Artificial Bee Colony (ABC) Algorithm," Springer Journal of Global Optimization, No. 39, 2007, pp. 459-471.
- [13] Karaboğa, D. and Başturk, B., "On the Performance of Artificial Bee Colony (ABC) Algorithm," Elsevier Journal of Applied Soft Computing, Vol. 8, 2008, pp. 687-698.
- [14] Karaboğa, D. and Akay, B., "A Comparative Study of Artificial Bee Colony Algorithm," Elsevier Journal of Applied Mathematics and Computation, Vol. 214, 2009, pp. 108-132.
- [15] Karaboğa, D. and Akay, B., "A Modified Artificial Bee Colony (ABC) Algorithm for Constrained Optimization Problems," Elsevier Journal of Applied Soft Computing, Vol. 11, 2011, pp. 3021-3031.
- [16] Akay, B. and Karaboğa, D., "A Modified Artificial Bee Colony Algorithm for Real-Parameter Optimization," Elsevier Journal of Information Sciences, Vol. 192, 2012, pp. 120-142.
- [17] Hagraş, H., "A Hierarchical Type-2 Fuzzy Logic Control Architecture for Autonomous Mobile Robots," IEEE Transactions of Fuzzy Systems, Vol. 12, No. 4, 2004, pp. 524-539.
- [18] Astudillo, L. and et al., "Intelligent Control of an Autonomous Mobile Robot using Type-2 Fuzzy Logic," Engineering Letters from Advanced Online Publications for International Associations of Engineers, Vol. 13, Issue 2, 2006, pp. 88-93.
- [19] Martinez, R. and et al., "Optimization with Genetic Algorithms of Interval Type-2 Fuzzy Logic Controllers for an Autonomous Wheeled Mobile Robot: A Comparison under Different Kinds of Perturbations," IEEE International Conference on Fuzzy Systems, Hong Kong, 2008, pp. 901-908.
- [20] Martinez, R. and et al., "Optimization of Interval Type-2 Fuzzy Logic Controllers for a Perturbed Autonomous Wheeled Mobile Robot using Genetic Algorithms," Elsevier Journal of Information Sciences, No. 179, 2009, pp. 2158-2174.
- [21] Leottau, L. and Melgarejo, M., "A Simple Approach for Designing a Type-2 Fuzzy Controller for a Mobile Robot Application," IEEE Annual Meeting of the North American Fuzzy Information Processing Society, Toronto, Canada, 2010, pp. 1-6.
- [22] Junratanasiri, S. and et al., "Navigation System of Mobile Robot in an Uncertain Environment Using Type-2 Fuzzy Modeling," IEEE International Conference on Fuzzy Systems, Taipei, Taiwan, 2011, pp. 1171-1178.
- [23] Karr, C. L., "Design of an Adaptive Fuzzy Logic Controller Using a Genetic Algorithm," Proceedings of the IEEE International Conference on Genetic Algorithms, 1991, pp. 450-457.
- [24] Shi, Y. and Eberhart, R., "Fuzzy Adaptive Particle Swarm Optimization," Proceedings of the IEEE International Congress of Evolutionary Computations, Seoul, Korea, Vol. 1, 2001, pp. 101-106.
- [25] Tian, D. and Li, N., "Fuzzy Particle Swarm Optimization Algorithm," IEEE International Joint Conference on Artificial Intelligence, Hainan Island, China, 2009, pp. 263-267.
- [26] Wong, C. and et al., "PSO-based Motion Fuzzy Controller Design for Mobile Robots," Taylor & Francis International Journal of Fuzzy Systems, Vol. 10, No. 1, 2008, pp. 284-292.
- [27] Valdez, F. and et al., "Fuzzy Control of Parameters to Dynamically Adapt the PSO and GA Algorithms," IEEE International Conference on Fuzzy Systems, Barcelona, Spain, 2010, pp. 1-8.
- [28] Martinez, R. and et al., "Optimization of Type-2 Fuzzy Logic Controllers for Mobile Robots Using Evolutionary Methods," Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics, San Antonio, Texas, USA, 2009, pp. 4764-4769.
- [29] Castillo, O. and et al., "Comparative Study of Bio-inspired Algorithms Applied to the Optimization of Type-1 and Type-2 Fuzzy Controllers for an Autonomous Mobile Robot," Elsevier Journal of Information Science, Vol. 192, 2012, pp. 19-38.
- [30] Abedinia, O. and et al., "Robust Fuzzy PSS Design using ABC," IEEE 10th International Conference on Environment and Electrical Engineering, Rome, Italy, 2011, pp. 1-4.
- [31] Bhattacharjee, P. and et al., "Multi-Robot Path-Planning Using Artificial Bee Colony Optimization Algorithm,"

- IEEE 3rd World Congress on Nature and Biologically Inspired Computing, Salamanca, Spain, 2011, pp. 219-224.
- [32] Wu, D., “Design and Analysis of Type-2 Fuzzy Logic Systems,” M.Sc. Thesis in Electrical and Computer Engineering, College of Engineering, National University of Singapore, 2005.
- [33] Wu, D. and Nie, M., “Comparison and Practical Implementation of Type-Reduction Algorithms for Type-2 Fuzzy Sets and Systems,” IEEE International Conference on Fuzzy Systems, Taipei, Taiwan, 2011, pp. 2131-2138.
- [34] Mondada, F. and et al, “The E-puck, a Robot Designed for Education in Engineering,” In Proceedings of the 9th Conference on Autonomous Robot Systems and Competitions, Castelo Branco, Portugal, Vol. 1, No. 1, 2009, pp. 59-65.
- [35] Michel, O. and et al, Cyberbotics’ Robot Curriculum, Cyberbotics Ltd., Wikibooks, Switzerland, 2010.