Robot Cooperation without Explicit Communication by Fuzzy Signatures and Decision Trees

Áron Ballagi[1] László T. Kóczy[2,3] Tamás D. Gedeon[4]

1 Department of Automation,
Széchenyi István University,
H-9026, Győr, Egyetem tér 1., Hungary

2 Inst. of Mechanical, Electrical Engineering and Information Technology,
Széchenyi István University,
H-9026, Győr, Egyetem tér 1., Hungary

3 Department of Telecommunications and Media Informatics,
Budapest University of Technology and Economics,
H-1117, Budapest, Magyar tudósok krt. 2., Hungary

4Department of Computer Science,
Australian National University,
Acton, ACT (Canberra), Australia

Email: ballagi@sze.hu, koczy@tmit.bme.hu, tom.gedeon@anu.edu.au

Abstract*—This paper presents a novel action selection method for multi robot task sharing problem. Two autonomous mobile robots try to cooperate for push a box to a goal position. Both robots equipped with object and goal sensing, but do not have explicit communication ability. We explore the use of fuzzy signatures and decision making system to intention guessing and efficient action selection. Virtual reality simulation is used to build and test our proposed algorithm.*

Keywords— Mobile robots, cooperation, fuzzy signatures, action selection, intention guessing.

1 Introduction

In this paper we focus on manipulation tasks with sufficiently challenging dynamics to require the careful cooperation of two or more robots. Sensing and actuation is noisy and uncertain in mobile robot domains, resulting in partial knowledge about the world. We explore the use of fuzzy signatures to efficient action selection and intention guessing in this environment. The intention guessing is the base of a meta-communication method between the robots. In this setup is not any explicit communication line.

We chose box-pushing as the problem domain because it has both theoretical interest and practical applications as it is an instance in large class of practical object manipulation tasks that appear to require tight feedback and control of real-world physics and dynamics [1-9]. From a theoretical standpoint, box-pushing is a variant on canonical object manipulation problem that draws on issue in fine motion as well as high level planning and control. Box-pushing is related to the well-known “piano-movers problem” [10], in that it requires the achievement of top level goal of delivering the box to a particular location, as well as the maintenance of low-level requirements including obstacle avoidance, maintaining contact with the box, and maintaining forward motion. From a practical point of view, box-pushing is a prototypical problem for studying various tasks requiring cooperation of number of smaller robots moving larger objects [6].

We use simulation in our experiments where two robots push a box to a goal position. Each robot has the own behaviour based control system thus they are fully autonomous. An action selection mechanism works in our behaviour based control in which the decision about selection is done by fuzzy signature based state describing algorithm.

2 Experimental task and environment

The actual stage of our research we use simulation of our real differential driven autonomous micro-robots (Fig. 1). The physical simulation is exact model of our robots in the case of scale, weight, mechanical systems and sensors.

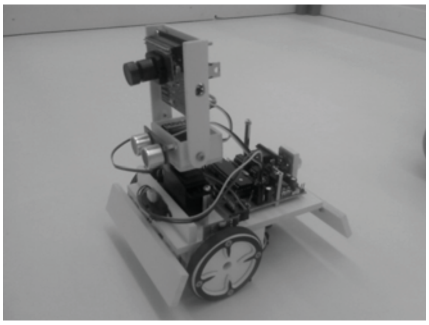


Figure 1. The real box-pushing robot

The two robots operate in a 2 x 2 m square arena. The box to be pushed is a 20 cm high and 40 cm wide and long. The

ISBN: 978-989-95079-6-8

goal region, located in one corner of the arena (Fig. 2), indicated by light sources detectable by the robot's sensor. In the experiments described here the goal location is fixed but it can be moved before and during experiments.

Each robot pushes the box with two "whiskers" which have a pair of force sensors. The left and right whiskers each provides an analog force signal which is combined to give information the relative position to the box and via the control loop keep it contact on both sides and thus perpendicular to the box.

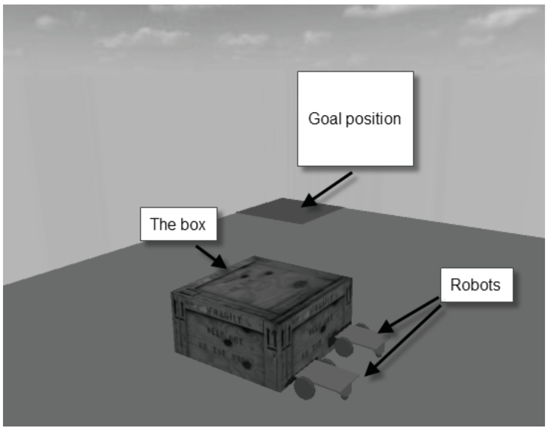


Figure 2. The simulation environment

In the description two direction sign systems are used, the absolute direction with letters N, E, S, W as in the usual sense for North, East, South and West. The second direction sign system is a box relative system where the sides of the box are N_B, E_B, S_B and W_B respectively (Fig. 3). The position of the objects (boxes and robots in this case) always can be described by the absolute course, latitude and longitude of the object. One object relative position to a box is described by the box relative system, i.e. which side of the box is touched by that object. For simplicity we assume that the sides of the box are always parallel with the N-S and E-W axes, so there is not necessary any rotation.

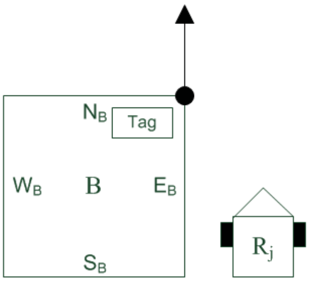


Figure 3. Symbols of boxes and robots

There are just a few essentially different robot positions allowed. Because two robots are needed for pushing the box, at each side of the boxes, two spaces are available for the robots manipulating them: the "counterclockwise position" and the "clockwise position" (see Fig. 4). The position is described by $P_r = [S, T]$ where *r* is the number of the robot, *S* is the side of the box where the robot touch it (N_B, E_B, S_B and W_B respectively) and the *T* is the turning position that means "counterclockwise position" or "clockwise position" (CC or CW).

The cooperating combination of robots is denoted by $C_{i,j}$ where *i,j* is the number of the robots. $C_{1,2} = P$ is the "pushing or shifting combination", when two robots (R_1 and R_2) are side by side at the same side of the box as Fig. 5 shows. In this case R_1 and R_2 are in the relative North (N_B) position. Of course, all the other three directions are similarly allowed. Any other combination of two robots is illegal, except see the next paragraph ("stopping combination").

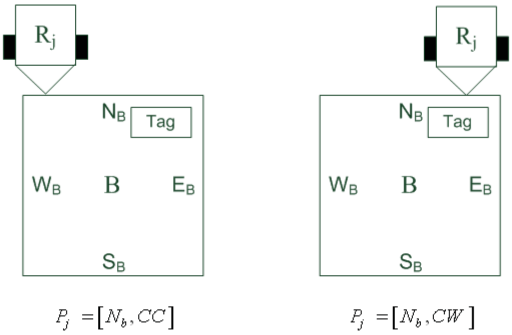


Figure 4. Robot positions at the N_B side of the box

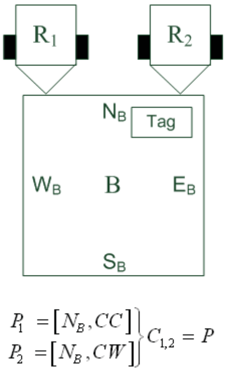


Figure 5. Allowed combinations of two robots for moving the box

Eventually, in Fig. 6, the combinations are shown where one robot intends to do a move operation, and another robot that has recognized the goal box configuration positions itself to prevent a certain move. This is an exception where a two robot combination other than the ones listed in Fig. 5 is legal as a temporary combination, clearly signalizing "stop this attempt as it is in contrary to the goal ".

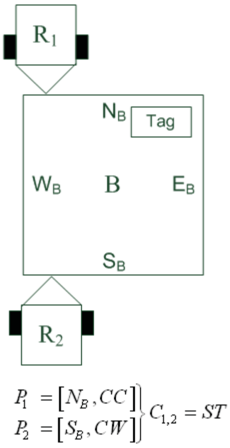


Figure 6. The stopping combination

3 Build up the codebook

After having overviewed the possible positions and combinations, let us build the part of the codebook that enables robots to recognize a situation and take action accordingly. In initial position both robots can see the goal light and know the actual position of the box. There are four good position and two pushing combination that the robots can take up.

How can take the starting position? The possible initial actions are:

1. The both robots move to reach the nearest possible good position.
2. Which reach the position first wins the temporal leader role if the other knows or guesses that this position is well for forming a useful combination. In this case the second robot turns and moves to the free position for this combination as the Fig. 7 shows. If the second robot rejects this combination then they try to take up a new form.
3. After the forming this starting combination the next is the push or shift task. We assume there are only two shifting axis, the N-S and E-S. The robots push the box a given force.

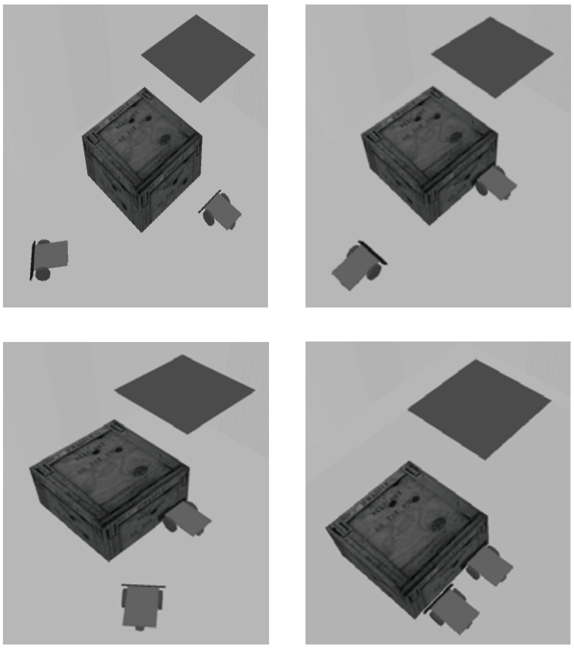


Figure 7. The robots take the starting combination

From this point a lot of scenarios are possible. Let us see some examples.

The R2 robot loses the goal sign, so it does not know the direction exactly. The R2 will slow down. The R1 robot detects the rotation of the box via its force sensors and guesses the R2 is slacken. The R1 has to make a decision, in other words it selects an action or reaction. There are two ways:

1. The R1 knows the right direction and forces to shift toward. It speeds up slightly, the R2 senses this and has to decide its reaction. The R2 makes a second self-examination and if it has not any other problem but the lost of goal sign then it switches to the *blind push* action or behavior. It means R2 pushes the box lean on R1 as follower. If R2 finds any other reason of slacken, e.g.: external obstacle or any internal error (e.g.: low battery), it keeps its own speed or decreases it. In this case the R1 gives a new reaction and so on, as long as they reach a right deal or give up the task.
2. The R1 does not sure the right direction, slow down to the speed of R2 and both robots search the goal light. If one of they find the goal takes the leader role and forces the other to push the box. If not, after a given time they stop pushing the box and move to take a new combination. After the some new unsuccessful attempt they give up.

Based on the above example and considerations it is possible to build up some elements of the action selection algorithm as a codebook. It will take the form of a decision tree, where the inputs are the direct observation, the first level outputs are intention guesses and the second level outputs the concrete actions of the corresponding robot.

Now let us see a relevant part of the codebook contains a decision tree with fuzzy elements. It is a part of the above presented example.

```
In R1 control system:
  Does R2 {slow down}?
    If  no then No Action
    else
    Do I know the {goal position}?
      If no then Goal Searching Action
      else
      Force Move Action
    Does R2 {accelerate}?
      If no then Slow Down Action
      else
      No Action

In R2 control system:
  Can I see the {goal light}?
    If  no then Slow Down Action
    .
    .
    .
  Does R1 {force the move}?
      If no then No Action (or Goal Searching Action)
      else
      Do I have any internal or external {obstacle}?
    If no then Blind Push Action
    else
    Slow Down Action (or Keep Move Action)
```

Note that in the condition parts there are fuzzy notions which are between curly brackets. It is usually hard to judge that the R2 slows down really or the box hits an obstacle as the information come from some simple sensors that might provide only approximate results.

This simple example illustrates clearly that the meta-communication among intelligent robots by intention guessing and fuzzy evaluation of the situation might lead to effective cooperation and the achievement of task that cannot be done without collaboration and communication.

4 Fuzzy signatures and decisions

4.1 *Fuzzy signatures*

In 1999 Vámos, *et al.* introduced the concept of Fuzzy Signatures [11]. Some further advanced versions of the concept and it possible use for describing complex data were later proposed in [12,13,14].

The original definition of fuzzy sets was $A : X \rightarrow [0,1]$, and was soon extended to *L-fuzzy sets* by Goguen [15]

$$A_s : X \rightarrow [a_i]_{i=1}^{k}, a_i = \left\{ \frac{[0,1]}{[a_{ij}]_{j=1}^{k_i}}, a_{ij} = \left\{ \frac{[0,1]}{[a_{ijl}]_{l=1}^{k_{ij}}} \right. \right. \tag{1}$$

$A_L : X \rightarrow L$, L being an arbitrary algebraic lattice. A practical special case, *Vector Valued Fuzzy Sets* was introduced by Kóczy [16], where $A_{V,k} : X \rightarrow [0,1]^k$, and the range of membership values was the lattice of *k*-dimensional vectors with components in the unit interval. A further generalization of this concept is the introduction of fuzzy signature and signature sets, where each vector component is possibly another nested vector (right).

Fuzzy signature can be considered as special multidimensional fuzzy data. Some of the dimensions are interrelated in the sense that they form sub-groups of variables, which jointly determine some feature on higher level. Let us consider an example. Fig. 8 shows a fuzzy signature structure.

The fuzzy signature structure shown in Fig. 8 can be represented in vector form as follow:

$$x = \begin{bmatrix} \begin{bmatrix} x_{11} \\ x_{12} \end{bmatrix} \\ \begin{bmatrix} x_{21} \\ \begin{bmatrix} x_{221} \\ x_{222} \\ x_{223} \end{bmatrix} \\ x_{23} \end{bmatrix} \\ \begin{bmatrix} x_{31} \\ x_{32} \end{bmatrix} \end{bmatrix}^T \tag{2}$$

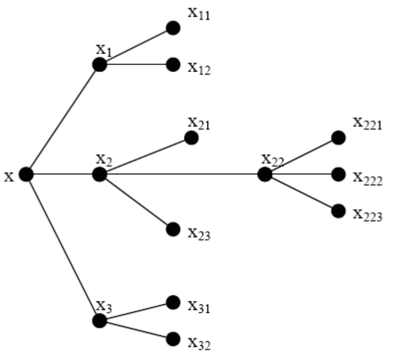


Figure 8. A Fuzzy Signature Structure

Here [x_{11} x_{12}] from a sub-group that corresponds to a higher level compound variable of x_1. [x_{221} x_{222} x_{223}] will then combine together to form x_{22} and [x_{21} [x_{221} x_{222} x_{223}] x_{23}] is equivalent on higher level with [x_{21} x_{22} x_{23}] = x_2. Finally, the fuzzy signature structure will become x = [x_{221} x_{222} x_{223}] in the example.

The relationship between higher and lower level is govern by the set of fuzzy aggregations. The results of the parent signature at each level are computed from their branches with appropriate aggregation of their child signature. Let a_1 be the aggregating associating x_{11} and x_{12} used to derive x_1, thus $x_1 = x_{11}a_1x_{12}$. By referring to Fig. 8, the aggregations for the whole signature structure would be a_1, a_2, a_{22} and a_3. The aggregations a_1, a_2, a_{22} and a_3 are not necessarily identical or different. The simplest case for a_{22} might be the min operation, the most well known t-norm. Let all aggregation be min except a_{22} be the averaging aggregation. We will show the operation based on the following fuzzy signature values for the structure in the example.

Each of these signatures contains information relevant to the particular data point x_0; by going higher in the signature structure, less information will be kept. In some operations it is necessary to reduce and aggregate information obtained from another source (some detail variables missing or simply being locally omitted). Such is when interpolation within a fuzzy signature rule base is done, where the fuzzy signature flanking an observation are not exactly of the same structure. In this case the maximal common sub-tree must be determined and all signatures must be reduced to that level in order to be able to interpolate between the corresponding branches or roots in some cases [17].

$$x = \begin{bmatrix} \begin{bmatrix} 0.3 \\ 0.4 \end{bmatrix} \\ \begin{bmatrix} 0.2 \\ \begin{bmatrix} 0.6 \\ 0.8 \\ 0.1 \end{bmatrix} \\ 0.9 \end{bmatrix} \\ \begin{bmatrix} 0.1 \\ 0.7 \end{bmatrix} \end{bmatrix}^T \tag{3}$$

After the aggregation operation is perform to the lowest branch of the structure, it will be described on higher level as:

$$x = \begin{bmatrix} 0.3 \\ \begin{bmatrix} 0.2 \\ 0.5 \\ 0.9 \end{bmatrix} \\ 0.1 \end{bmatrix}^T \tag{4}$$

Finally, the fuzzy signature structure will be:

$$x = \begin{bmatrix} 0.3 \\ 0.2 \\ 0.1 \end{bmatrix}^T \qquad (5)$$

4.2 Fuzzy signatures for box-pushing robot cooperation

Now, let us construct the fuzzy signatures for robot cooperation. The fuzzy signatures presented here are implemented in R1 robot action selection system as a part of the meta-communication codebook. Of course here is described only a slice of the whole signatures.

Fig. 9 presents the signatures which describe R2 behaviour. The information communicated to R1 is partly the observation about the last move of R2 and partly the evaluation of the situation with the box, according to those membership degrees will be attached to each leave of the actual signature.

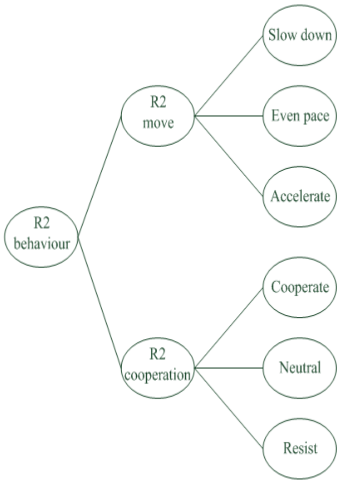


Figure 9. R2 robot's behavior fuzzy signatures

Figure 11 presents a portion of decision tree applied by R1 when it observes some action or reaction of R2 described by above mentioned behavior fuzzy signatures.

After all action selection the R1 control system reevaluates the values of the membership degrees of observed R2 behavior, environmental properties and the own state as the lines partially show in Fig. 11. After the evaluation the robot makes a decision which action is taken.

Parallel, in the R2 control system runs a similar task which reacts to the R1 decision and action. Thus, the two robots produce a circle of action and reaction (Fig. 10), where an initial action triggers the circulation of reactions and composes a meta-communication between the robots.

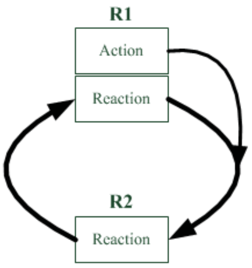
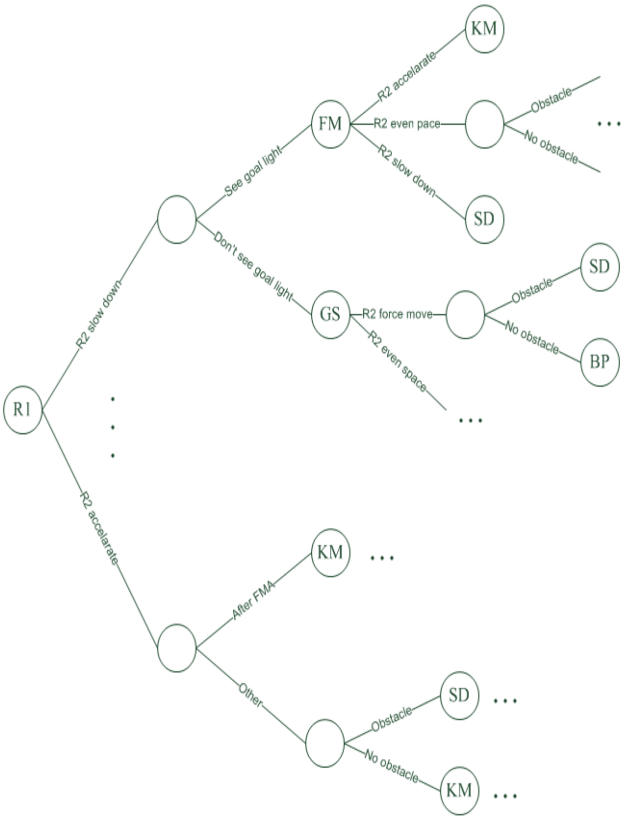


Figure 10. Action – reaction circle

This is a type of context dependent or fuzzy communication [18, 19], which means every robot has the own codebook and communicate in a vague, compressed or quasi channel.

Then the robots build up and interpret this imprecise information with their codebooks.



Where the actions are

- BP – Blind push
- FM – Force move
- GS – Goal Searching
- KM – Keep move
- SD – Slow down

Figure 11. The decision tree of R1 robot

5 Conclusions

Fuzzy communication contains vague or imprecise components and it might lack abundant information. If two robots are communicating by a fuzzy channel, it is necessary that both ends possess an identical part within the codebook. The codebook might partly consist of common knowledge but it usually requires a context dependent part that is learned by communicating. Possibly it is continuously adapting to the input information. If such a codebook is not available or it contains too imprecise information, the information to be transmitted might be too much distorted and might lead to misunderstanding, misinterpretation and serious damage. If however the quality of the available codebook is satisfactory, the communication will be efficient i.e. the original contents of the message can be reconstructed. At the same time it is cost effective, as fuzzy communication is compressed as compared to traditional communication. This advantage can be deployed in many areas of engineering, especially where the use of the communication channels is expensive in some sense, or where there is no proper communication channel available at all.

Here we illustrate clearly that the communication among intelligent robots by intention guessing and fuzzy evaluation of the situation might lead to effective cooperation and the achievement of tasks that cannot be done without collaboration and communication.

We simulated many scenarios and almost got acceptable results, but sometimes the robots made deadlock combination and gave up the work. In future we want to work out new algorithms to solve these situations.

Acknowledgment

The research was supported by HUNOROB project (HU0045), a grant from Iceland, Liechtenstein and Norway through the EEA Financial Mechanism and the Hungarian National Development Agency, the Széchenyi University Main Research Direction Grant 2009, and a National Scientific Research Fund Grant OTKA T048832 and K75711.

References

[1] P. Caloud, W. Choi, J. C. Latombe, C. L. Pape, and M. Yim, Indoor automation with many mobile robots, *in IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 67-72, 1990.

[2] B. Donald, J. Jennings, and D. Rus, Analyzing teams of cooperating mobile robots, *in IEEE International Conference on Robotics and Automation,* pp. 1896-1903, 1994.

[3] C. R. Kube and H. Zhang, Collective robotic intelligence, *in Second International Confernece on Simulationof Adaptive Behavior: From Animals to Animats*, pp. 460-468, MIT Press, (Cambridge, MA), 1992.

[4] C. R. Kube and H. Zhang, Task modelling in collective robotics, *Autonomous Robots* 4(1), pp. 53-72, 1997.

[5] C. R. Kube and E. Bonabeau, Cooperative transport by ants and robots, *Tech. Rep.* 99{01-008, Santa Fe Institute, Santa Fe, NM, 1999.

[6] M. J. Matari_c, M. Nilsson, and K. T. Simsarian, Cooperative multi-robot box-pushing, *in IEEE International Conference on Intelligent Robots and Systems*, vol. 3, pp. 556-561, 1995.

[7] F. R. Noreils, An architecture for cooperative and autonomous mobile robots, *in IEEE International Conference on Robotics and Automation*, pp. 2703-2710, 1992.

[8] L. E. Parker, ALLIANCE: An architecture for fault tolerant, cooperative control of heterogeneous mobile robots, *in IEEE/RSJ/GI International Conference on Intelligent Robots and Systems*, pp. 776-783, 1994.

[9] D. J. Stilwell and J. S. Bay, Toward the development of a material transport system using swarms of ant-like robots, *in IEEE International Conference on Robotics and Automation*, pp. 766-771, 1993.

[10] J. Schwartz and M. Sharir, On the Piano Mover’s problem. II. General techniques for computing topological properties of real algebraic manifolds’, *Advances in Applied Math*, 1983.

[11] T. Vámos, Gy. Bíró and L. T. Kóczy, Fuzzy signatures, *in SIC-EUROFUSE'99,* Budapest, pp. 210-217, 1999.

[12] B. S. U. Mendis, T. D. Gedeon and L. T. Kóczy, Investigation of Aggregation in Fuzzy Signatures, *in 3rd International Conference on Computational Intelligence, Robotics and Autonomous Systems*, (CD proc), Singapore, 2005.

[13] K. W. Wong, T. D. Gedeon and L. T. Kóczy, Construction of fuzzy signature from data, *in FUZZ-IEEE 2004*, Budapest, pp. 1649-1654, 2004.

[14] K. W. Wong, A Chong, T. D. Gedeon, L. T. Kóczy, and T. Vámos, Hierarchical fuzzy signature structure for complex structured data, *in International Symposium on Computational Intelligence and Intelligent Informatics (ISCIII),* Nabeul, pp. 105-109, 2003.

[15] J. A. Goguen, L-fuzzy Sets, *J. Math. Anal. Appl.* 18 pp. 145-174, 1967.

[16] L. T. Kóczy, Vectorial I-fuzzy Sets, *in M. M. Gupta & E. Sanchez (eds.): Approximate Reasoning in Decision Analysis*, North Holland, Amsterdam, pp. 151-156, 1982.

[17] T. Vámos, L.T. Kóczy and Gy. Biró, Fuzzy Signatures in Data Mining, *in IFSA World Congress and 20th NAFIPS International Conference*, Vancouver, pp. 2842-2846, 2001.

[18] L.T. Kóczy and T.D. Gedeon, Context Dependent Reconstructive Communication, *3rd International Symposium on Computational Intelligence and Intelligent Informatics*, ISCIII 2007, Agadir, Morocco, March 28-30, pp. 13-19, 2007.

[19] A. Ballagi, L.T. Kóczy and T.D. Gedeon, Local Codebook Construction for Fuzzy Communication in Cooperation of Mobile Robots, *in First Győr Symposium on Computational Intelligence*, Győr, 2008.