Dynamic Autonomous Agent Positioning Based on Computational Intelligence

Mohammad Mehdi Korjani, Ahmad Afshar, Mehrab Norouzitallab, Mohammad Bagher Menhaj

***Abstract*—In this paper, we present a modular neural network for learning formation strategy in multi-agent systems. A supervised learning method is devised to train the modular neural network in order for a group of agents to learn formation strategy in an environment. At first, the environment conditions are separated into some different parts called contexts in this paper. Consequently, each agent employs a neural network to learn the sequence of actions of expert, according to the present context. After the training process, agents would be able to imitate human behavior in similar conditions. As a result, an intelligent model of human behavior is extracted which contributes in building autonomous agents. This framework increases the robustness and efficiency of the multi-agent system while providing the system with redundancy, reconfiguration ability and structure flexibility.**

The fuzzy ARTMAP neural network combines a unique set of computational abilities that are needed to function autonomously in a changing world. These Characters lead us to use this network in learning process. Therefore, the modular fuzzy ARTMAP neural network is used to extract expert knowledge in formation strategy. In particular, the proposed framework is applied to soccer robots and its generalization capability is evaluated with datasets from which several data points are randomly removed.

***Index Terms*—Formation Strategy, Learning Behavior, Modular Neural Network, Multiagent Systems.**

I. INTRODUCTION

In the recent years formation strategy in multi-agent systems has attracted considerable attention [1-8]. These include satellite and spacecraft, underwater vehicles, drone planes and capturing/enclosing an invader [9-17].

There are two main approaches to formation in multi-agent systems namely model based [18] and behavior based formation [19]. Many methods are discussed in each field, for example, leader-follower strategy [20], virtual structure approach [21] and behavior-based method [22]. In model based formation strategy, the exact model of the robot, task and environment are built. Despite the time required to develop such a model, the operation of this model is limited in unknown environments. In contrast, in behavior based modeling, the exact models are not necessary. At each behavior, particular goal is assumed and entire task can be covered by all behaviors. In the Leader-follower research area, a lead vehicle that is either manually driven or autonomous is followed by a series of automated robots [23], [24]. The main criticism to the leader-follower approach is that the formation does not tolerate leader faults and exhibits poor disturbance rejection features. In spite of these deficiencies the leader-follower approach is particularly appreciated because of its simplicity and scalability.

Robot formation in the virtual structure approach [25] is considered as a single virtual rigid structure. In this approach, the behavior of the robotic system is assimilable to that of a physical object. Desired trajectories are not assigned to each single robot but to the entire formation as a whole. In this case the behavior of the robot formation is predictable and consequently the control of the robot formation is straightforward. Nevertheless a large inter-robot communication bandwidth is required.

In the behavior based approach [26], [27] some behaviors are prescribed to each robot. The relative importance of each behavior specifies the robot final action. The theoretical formalization and mathematical analysis of this approach is difficult and consequently it is not easy to guarantee the convergence of the formation to a desired configuration.

One of the main goals in formation strategy is to achieve a formation while using only information of positions. The objective investigated in this paper is that of attaining a dynamic formation. This means conditions are assumed such as a resource shortage, a monitoring target moving in environment and strategy is changed based on situations. Each agent has a partially information about the environment. The goal of agents is to achieve and maintain the pre-defined positions to the leader which may change in different situation. Different context change the strategy of the formation.

In this study, we will introduce how formation strategy as high level behaviors can be modeled by extracting knowledge of an expert. This framework can increase the performance of agents' team formation strategy in dynamic environments. As a case study, we use soccer simulation environment in which

Manuscript received January 7, 2009. (M. M. Korjani is with the Electrical Engineering Department, Amirkabir University of Technology, 424 Hafez Ave., Tehran, Iran, (Tel: +98-21-64543391; Fax: +98-21-6406469; e-mail: Korjani@ieee.org).

A. Afshar is with the Electrical Engineering Department, Amirkabir University of Technology, (e-mail: aafshar@aut.ac.ir).

M. Norouzitallab is with Computer Engineering Department, Amirkabir University of Technology, (e-mail: norouzitallab@aut.ac.ir).

M. B. Menhaj is with the Electrical Engineering Department, Amirkabir University of Technology, (e-mail: tmenhaj@ieee.org).

formation strategy can be considered by a team of eleven autonomous robots. In this dynamic environment, robots (the slaves) track the position of the ball (the master) as a reference point in the environment. Agents learn high level behaviors based on the environment conditions and expert knowledge.

At first, a task and the goal of the task are defined to be implemented by agents. The expert determines the strategy of obtaining the goal. Then he divides the environment conditions into some parts which are referred to as contexts in this paper. After defining contexts, the expert considers a high level behavior that must be implemented in each context to obtain the goal of the task. The term "high level behaviors" is used here because they are behaviors that involve some simple behaviors which are called low level behaviors [28-30]. It is furthermore important to notice that in contrast to high level behaviors, low level behaviors do not need learning algorithms to be implemented.

In the training step, the expert takes some actions to complete the task. In this part, first, the context of the environment is inferred and then Fuzzy ARTMAP as a knowledge based neural network is utilized to extract knowledge from expert behaviors.

The process of extracting knowledge from expert action and context based reasoning [31] in learning algorithm in multi-agent formation strategy has some advantages as:

After execution of low level behaviors, the time required to develop a model of high level behaviors could be significantly reduced. The only time consuming part is execution of low level behavior.

This method allows agents to learn knowledge form either unwilling experts or experts who are unable to explicit their knowledge to a third party [32].

The idea of using context based reasoning is to decrease the complexity of fuzzy ARTMAP models when used in applications with a large number of training patterns. We can overcome to disadvantage of Fuzzy ARTMAP-based models which is sensitivity to noise. This can cause category proliferation during learning and misclassification during recalling.

$$\Gamma(O_{T-t_0}) = \Lambda_{c_i} = \{\delta_1, \delta_2, ..., \delta_n\} \qquad (3)$$

This system which has learned high level behaviors is an intelligent model of expert behavior for accomplishment of a formation task. We have conducted experiments proving that by using this framework, agents can be efficiently positioned. We used the RoboCup Soccer 2D Simulator [33], [34] as the experimental environment.

The following sections provide some background information on theory and terminology used in this paper. In section two, a brief introduction to extract knowledge from observation will be introduced. Context based reasoning is defined in section three. Fuzzy ARTMAP as a knowledge based neural network will be discussed in section four. Simulation results will be presented in section five. In final section some conclusions and future research on the proposed framework are considered.

II. Extracting knowledge from observation

The common thing in all machine learning techniques is that data is gathered from real world. First knowledge beyond the manipulated data must be extracted. The method developed here can extract knowledge from observing high level behaviors of an expert in a multi-agent system. This method is employed to learn formation behavior from observing expert actions.

Learning from observation associated with unsupervised learning was first mentioned in Michalski, Carbonel and Mitchell's book [35].Although, with this view point, training data could be obtained by observation but most of works are limited to learning low level behavior [36], [37].

Observation is a discrete point of time-dependent conditions that can be used to infer assertions about agents' environment. In this paper, observation in time t contains the characteristics of the environment.

$$O_t = < U_1, U_2, ..., U_n > \qquad (1)$$

where U_i is the any character of environment that shall be assumed in a context. In the simulation result, it is natural to consider that the ball is the most important focal target and its location is an important state variable, therefore, in this paper $O_t = < Ball\ Position >$.

At each context, Fuzzy ARTMAP as a knowledge-based neural network extracts knowledge from the observation by creating a mapping between the observation pattern and the observed response. In this study, the observation pattern is ball position and the observed response is the agent position.

Assume that an expert performs a high level action Λ in a given task when context c_i occurs. This high level behavior contains a sequence of level behaviors that implement in each time step respectively.

$$\Lambda = \{\delta_1, \delta_2, ..., \delta_n\} \qquad 1 \le i \le n \qquad (2)$$

The learning process derives some function Γ on a given observation sequence O_t.

The learning algorithm designated by Γ operates on the observation sequence O_{T-t_0} and outputs a set of assertions Λ_{C_i} that describe the behavior that has been observed.

III. Context based reasoning

When a task is assigned to a team of agents, each agent must execute a sequence of action based on the environment conditions to obtain the goal of task. Context based reasoning is a method to reduce the complexity of this activity.

Context based reasoning is based on the idea that [38], [39]:

A context calls for a set of actions and procedures that properly address the current conditions.

As a task evolves, a transition to another set of actions and procedures may be required to address a new context.

What is likely to happen under the current context is limited and influenced by the current context itself.

As it can infer from context based reasoning idea, the task and context are central concepts of context based reasoning and must be defined.

A task is an abstraction defined within the model and assigned to agents prior to run-time. Included within a task are the goal and constraints that must be declared. The goal indicates the conditions determining the finalization of the task. The task's goal can be formally defined as a function of a set of environmental and physical conditions. We have:

$$GOAL = f\,(E_c, P_c) \qquad (4)$$

The union of the set of physical constraints (L_p), environmental constraints (L_e) and logistical constraints (L_l) construct the constraints of a task.

$$Constraint = \{L_p, L_e, L_l\} \qquad (5)$$

In [40], environment and physical conditions which propose a particular behavior are assumed to comprise a context. An expert divides the environment conditions to some contexts.

$$C = \{C1, C2, \ldots, C3\} \qquad (6)$$

At any given time, environmental conditions represent a unique context which is called in this paper as the active context (C_i). The active context induces a certain agent behavior specific to that context [39]. Based on these ideas, it is clear that agents use only a fraction of their knowledge at any given time.

IV. Fuzzy ARTMAP Neural Network

An autonomous agent must be able to learn about rare events with important consequences, even if such events are similar to many other events that have different consequences. Many traditional learning schemes use a form of slow learning that tends to average similar event occurrences. In contrast, fuzzy ARTMAP systems can rapidly learn rare events whose predictions differ from those of similar events [41]. Rare events typically occur in a non-stationary environment, such as a large database, in which event statistics may change rapidly and unexpectedly. Individual events may also occur with variable frequencies and durations, and arbitrarily large numbers of events may need to be processed. Each of these factors tends to destabilize the learning process within traditional algorithms. Fuzzy ARTMAP was developed in the early 1990's by Carpenter et al [42]. Readers must refer to [43] for further information.

Each fuzzy ARTMAP system includes a pair of fuzzy ART modules (FUZZY ART_a and FUZZY ART_b), as shown in Figure 1. The FUZZY ART_a and FUZZY ART_b modules within Fuzzy ARTMAP are responsible for generating pattern clusters that correspond to a certain pattern form. During supervised learning, FUZZY ART_a receives a stream a_i^p of input patterns and FUZZY ART_b receives a stream b_i^p of input patterns, where b_i is the correct prediction given a_i. These modules are linked by an associative learning network and an internal controller that ensures autonomous system operation in real time. The map field is then responsible for creating a many-to-one mapping between the templates within FUZZY ART_a and those within FUZZY ART_b.

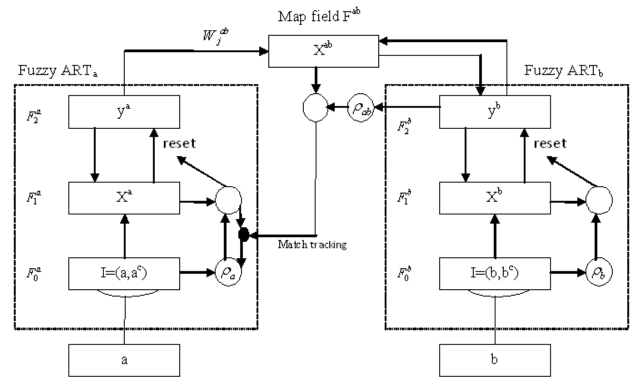

Fig 1: Fuzzy ARTMAP Architecture

Each fuzzy ART module (FUZZY ART_a, FUZZY ART_b) is composed of input, match and choice layers. Input layer (F_0) uses current input vector and complement input vector in order to prevent a category proliferation and transmits it to the next layer. At first, each input must be normalized and then the complement coded input is obtained by $a_i^c = 1 - a_i \,, a_i \in [0,1]$. Thus, at the input layer we have:

$$I = (a, a^c) = (a_1, \ldots, a_m, a_1^c, \ldots, a_m^c) \qquad (7)$$

where m is the length of the input pattern. Match layer (F_1) receives input (I) from the input layer (F_0) and top-down input from the choice layer (F_2) and matches the input pattern which has been clamped and the information stored in the top-down weights. Choice layer (F_2) represents a category of inputs grouped together around an exemplar generated during the self-organization of the fuzzy ART module. In this layer, the neurons are called category neurons.

The three steps of fuzzy ART learning process are choice, match and learning. In choice layer, once the complement coded input (I) is in the match layer, all category nodes become active to some degree and the winning node is chosen. This output activation is denoted $T_j(I)$ and is defined as:

$$T_j = \frac{|I \wedge W|}{\alpha + |W_j|} \qquad (8)$$

Here $(x \wedge y) = min(x, y)$ is the fuzzy AND operator [44] and α is generally a very low value. The highest output node is the winning category node. In this part, the winner-takes-all method is used.

After the winner category is computed, it will be analyzed to check if its similarity is higher than the minimum similarity allowed that is defined by the vigilance parameter(ρ). The

ISBN: 978-988-17012-2-0

vigilance parameter $\rho \in [0,1]$ calibrates to the minimum amount to correct the predictive error. The match function figures out the degree of the match between the category node and the input vector. This function is defined as:

$$\Gamma(I, W_i) = \frac{|I \wedge W_i|}{|I|} \quad (9)$$

If the match function is higher than the vigilance parameter ($\Gamma(I, W_i) > \rho$) then the winning node belongs to the same class and its weight vector W_i is updated to learn the new input pattern (learning phase). If the match function is higher than the vigilance, but the winning node does not represent the same class as that to which input vector belongs, then the match tracking process occurs. In this process, the vigilance parameter is increased in order to avoid a subsequent activation of the same category node. That is, a search for another category node is activated (choice phase). If the match function is lower than the vigilance, its activation output (T) is set to zero and a new search is activated (choice phase). The process of searching for a winner category neuron continues until a satisfactory node is found or a new node is assigned. Once both FUZZY ART modules produce their output, the map field model forms the association between both winning categories (from FUZZY ART_a and FUZZY ART_b).

Once the match conditions are satisfied, a resonance state is activated which allows learning to occur in the relevant section of the weight matrix. The learning equation is defined as follows.

$$W_j^{new} = \beta \cdot (I \wedge W_j^{old}) + (1-\beta) \cdot W_j^{old} \quad (10)$$

wherein β is the learning rate parameter ($0 \le \beta \le 1$). Fast learning corresponds to setting $\beta = 1$. After learning process is occurred, then FUZZY ART_a and FUZZY ART_b connect to each other via the map field module. Each category neuron in FUZZY ART_a module is linked to all the map field neurons (1-to-many association) and this weight is initially set to zero. During the learning phase, the weight linking the winner category neuron and its corresponding map field neuron is set to one. During the recalling phase, once the two fuzzy ART modules produce their output, an association between these outputs is employed. First, the winner node in the map field module is calculated, taking into account the output of the ART_a module, which can be defined as follows:

$$W(I) = Max\left(\sum_{k=1}^{N} U_k\right) \quad (11)$$

where

$$U_k = \sum_{j=1}^{M} w_{jk} T_j \quad (12)$$

W_{jk} is the weight from the j^{th} category neuron (F_2^a in Figure 1) to the k^{th} neuron of map field layer and T_j is the output of the j^{th} category neuron.

After the map field winner node is chosen, an association between the map field winner node and the FUZZY ART_b module winner node is created. Each map field neuron is linked only to its corresponding category FUZZY ART_b module neuron one-to-one association. Initially, the weights from the map field and FUZZY ART_b modules are set to zero. When an association occurs, the corresponding weight is set to one.

V. SIMULATION RESULTS

For simulation results, we use FormationEditor that was implemented by H. Akiyama et al. [45]. The main reasons to use this tool are that we can modify the data easily and train the agent with the presented data and then observe the agent's positions. Thereby, we developed this simulation environment to construct the proposed learning framework. Figure 2 shows the main screen of the GUI tool.

Fig 2: Screenshot of FormationEditor

We use two sets of agents' positions. In the first set, the training data is prepared by a human instructor who has no knowledge about soccer simulation (based on his intuition). In the second set, an expert in soccer simulation field has been told to prepare training data. In this set, agents' positions have been arranged more precisely.

The set of contexts contains *{Offence, Defense, Free Kick, Kick Off, Corner kick, Goal Kick}*. The target of the play is to reach to the desired position. Moreover, there are some constraints in the simulation environment. Maximum agent speed is an example of physical constraints. Readers may refer to [33] for further information.

Each agent partially observes the simulation field. Therefore, it must first find the its position and then consider the position of the ball and other agents. After it gathers the local information about the environment, the agent must conclude the desired position and implement some low level behaviors to reach to the goal.

The soccer field is 105 m × 68 m and the entire field is used. As shown in figure 3, first to implement the proposed learning algorithm, the position of each agent and ball is normalized to the interval [0, 1] and the context of learning algorithm is selected. Then the Fuzzy ARTMAP neural network is used to learn the structure of the data. In this paper 90 percent of the entire data is used for training and the rest of it is used for test. In the training phase, the position of each agent is fed to Fuzzy ART_a and the position of the ball recommended by the supervisor is fed to Fuzzy ART_b. After all training data are learned by the network, the test data is fed to network.

A standard three-layer feed-forward neural network which is widely used is considered here as a case to be compared with Fuzzy ARTMAP neural network. This network has sigmoid activation functions for each unit in hidden and output layers and connection weights between units are determined by back-propagation algorithm. This network is called BPN in this paper.

Simulation results show the average error and the maximum error between the recommended and the acquired position. To evaluate the generalization capability, some data are randomly removed from the original data set.

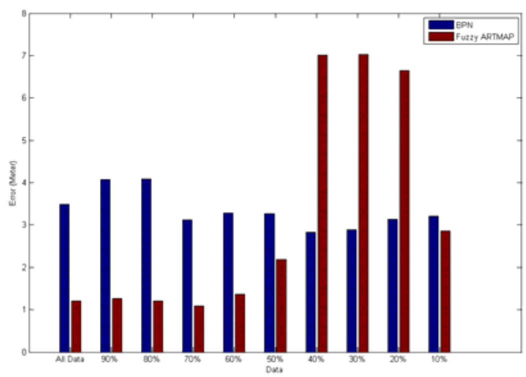

Fig 3: Average error BPN and Fuzzy ARTMAP (data set 1)

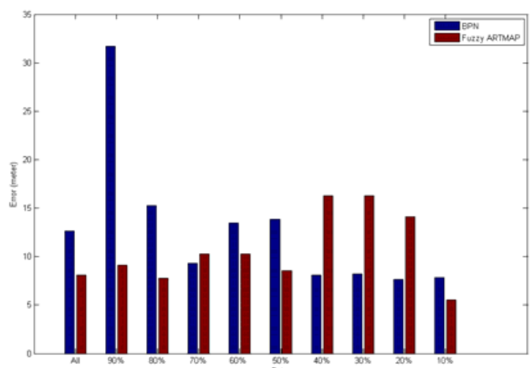

Fig 4: Maximum error BPN and Fuzzy ARTMAP (data set 1)

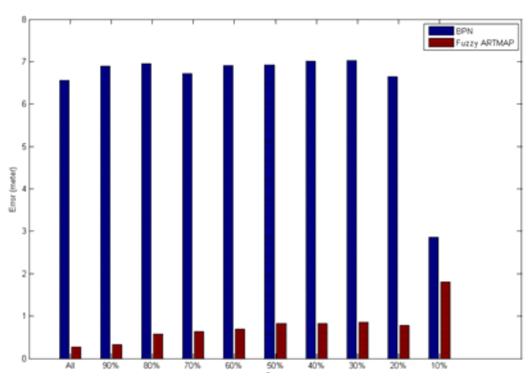

Fig 5: Average error BPN and Fuzzy ARTMAP (data set 2)

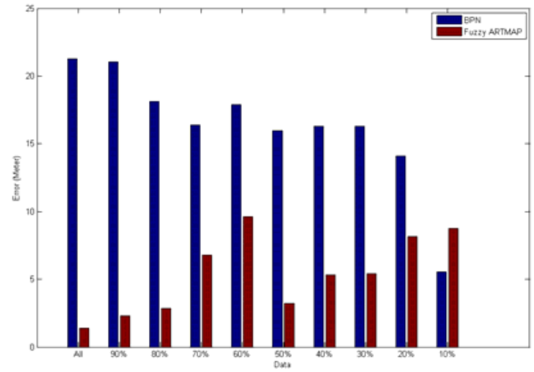

Fig 6: Maximum error BPN and Fuzzy ARTMAP (data set 2)

In figures 3-6, the average and maximum error value is illustrated versus the available percent of data. In data set 1, the average error and the maximum error of the proposed algorithm based on Fuzzy ARTMAP neural network is less than BPN neural network when 50 percent or more of the data is available. In data set 2, the results illustrate that if up to 90 percent of the data is randomly removed, the Fuzzy ARTMAP learning algorithm is more accurate. Thus, we can realize that the generalization capability of the proposed algorithm is very high and agents can operate more precisely. In contrast, the average error of the BPN is more than 10 times greater than that of Fuzzy ARTMAP. If the data are prepared precisely, the Fuzzy ARTMAP can learn the data more efficiently and faster than BPN neural network. Therefore, if a more complex and higher precision positioning is required, the fuzzy ARTMAP neural network must be employed.

VI. CONCLUSION

In this paper, we introduced a framework for multi agent formation strategy using learning from observation of expert behavior. Here fuzzy ARTMAP is used to extract expert behavior in a task, as a knowledge based neural network. Employing such a system helps us better understand expert behavior and obtain an intelligent model of expert behavior. Experimental tests were performed in order to investigate the benefits of using the proposed techniques within a multi agent system.

The main outcome of the experimental test is that the model proposed in this paper exhibited a higher performance than the conventional BPN. In addition, the experiments illustrate that the learning method aided by fuzzy ARTMAP converges more quickly than the conventional BPN. It also stabilizes with fewer training exemplars than the BPN and generates a good performance in this case.

ARTMAP-based model has a number of advantages over other model, such as knowledge extraction, no catastrophic forgetting and fast learning.

However, the main disadvantage of ARTMAP-based models is sensitivity to noise which can cause category proliferation during learning and misclassification during recalling.

Overcoming the main disadvantage of ARTMAP-based models was the main aim behind the proposal of the Context based reasoning model. The idea of using context based reasoning is to decrease the complexity of fuzzy ARTMAP models when used in applications with a large number of training patterns.

REFERENCES

[1] Y.Chen and Z. Wang, “Formation Control: a Review and a New Consideration”, IEEE/RSJ Conf. Robot. Syst., pp. 3181- 3186, 2005.

[2] R.W. Beard, J. Lawton, and F.Y. Hadaegh, “A Coordination Architecture for Spacecraft Formation Control”, IEEE Trans. Contr. Syst. Tech., Vol. 9, No. 6, pp. 777-790, 2001.

[3] H. G. Tanner, S. G. Loizou and K. J. Kyriakopoulos, “Nonholonomic Navigation and Control of Multiple Mobile Robot Manipulators”, IEEE Trans. Robot. Automat., Vol. 19, no. 1, pp. 53-64, 2003.

[4] R. Mead, J.B. Weinberg and J.R. Croxell, “An Implementation of Robot Formations using Local Interactions”, 22nd National Conference on Artificial Intelligence (AAAI-07), pp. 1989-90, 2007.

[5] D.V. Dimarogonas, M.M. Zavlanos, S.G. Loizou, K.J. Kyriakopoulos, “Decentralized motion control of multiple holonomic agents under input constraints”, Proc. of the 42nd IEEE Conference on Decision and Control, pp. 3390, 2003.

[6] F. Chen, Z. Chen, Z. Liu, L. Xiang and Z. Yuan, “Decentralized Formation Control of Mobile Agents: A Unified Framework”, Physica A 387, pp. 4917–4926, 2008.

[7] G. Lafferriere, A.Williams, J. Caughman and J.J.P. Veerman, “Decentralized Control of Vehicle Formations”, Systems and Control Letters 54, pp. 899 – 910, 2005.

[8] S. Monteiro and E. Bicho, “A Dynamical Systems Approach to Behavior-Based Formation Control”, International Conference on Robotics and Automation, 2002.

[9] H Yamaguchi, “A Cooperative Hunting Behavior by Mobile Robot Troops”, The International Journal of Robotics Research, Vol. 18, No. 8, pp. 931–940, 1999.

[10] N.E. Leonard and P. Ogren, “Obstacle Avoidance in Formation”, IEEE ICRA, pp. 2492–2497, 2003.

[11] W. Ren and R. Beard, “A Decentralized Scheme for Spacecraft Formation Flying via The Virtual Structure Approach”, AIAA J. Guidance, Control Dynamics Vol. 27, No. 1, pp. 73–82, 2004.

[12] A.G. Sparks, “Special Issue on Control of Satellite Formations”, Internat. J. Robust Nonlinear Control 12 , pp.2–3, 2002.

[13] H. Tanner, A. Jadbabaie and G. Pappas, “Stable Flocking of Mobile Agents, Part I: Fixed Topology”, in: Proceedings of IEEE Conference on Decision and Control, pp.2010–2015, 2003.

[14] H. Tanner, A. Jadbabaie and G. Pappas, “Stable Flocking of Mobile Agents, Part II: Dynamic Topology”, in: Proceedings of IEEE Conference on Decision and Control, pp. 2016–2021, 2003.

[15] Williams, S. Glavaski and T. Samad, “Formations of Formations: Hierarchy and Stability”, ACC2004, pp. 2992–2997, 2004.

[16] J.A. Fax and R.M. Murray, “Information Flow and Cooperative Control of Vehicle Formations”, IEEE Trans. Automat. Control, Vol. 49, No. 9, pp. 1465–1476, 2003.

[17] J.R. Lawton, R.W. Beard, and B.J. Young, “A Decentralized Approach to Formation Maneuvers”, IEEE Transactions on Robotic and Automation, vol. 19 No.6, pp.933–941, 2003.

[18] M Uchiyama and P Dauchez, “A Symmetric Hybrid Position/Force Control Scheme for the Coordination of Two Robots”, IEEE Int. Conf. Robotics and Automation, pp. 350–355, 1988.

[19] R. C. Arkin, “Behavior-Based Robotics”, MIT Press, Cambridge, 1998.

[20] N. Cowan, O. Shakerina, R. Vidal, and S. Sastry, “Vision-Based Followthe-Leader,” IEEE/RSJ International Conference on Intelligent Robots and Systems, Vol. 2, pp. 1796–1801, 2003.

[21] J. Ghommem, F. Mnmnif, G.Poisson and N. Derbel, “Nonlinear Formation Control of a Group of Underactuated Ships”, American Control conference, pp. 1746 – 1751, 2003.

[22] D. Scharf, F. Hadaegh, and S. Ploen, “A Survey of Space Formation Flying Guidance and Control (Part 2),” American Control Conference, Boston, Massachusetts, 2004.

[23] A.K. Das, R. Fierro, V. Kumar, J.P. Ostrowsky, J. Spletzer, and C. Taylor, “A Vision-Based Formation Control Framework”, IEEE Transaction on Robotics and Automation, Vol. 18, No.5, pp. 813–825, 2002.

[24] R. Vidal, O. Shakernia, and S. Sastry, “Following the Flock: Distributed Formation Control with Omnidirectional Vision-Based Motion Segmentation and Visual Servoing”, IEEE Robotics and Automation Magazine, Vol. 11, No. 4, pp.14–20, 2004.

[25] K.H. Tan and M.A. Lewis, “High Precision Formation Control of Mobile Robots Using Virtual Structures”, Autonomous Robots, Vol. 4 No. 4, pp. 387–403, 1997.

[26] T. Balch and R.C. Arkin, “Behavior-Based Formation Control for Multi-Robot teams”, IEEE Transactions on Robotics and Automation, Vol. 14, No.6, pp.926–939, 1998.

[27] N.E. Leonard and E. Fiorelli, “Virtual leaders, artificial potentials and coordinated control of groups”, In Proc. 40th IEEE Conference on Decision and Control, pp. 2968-2973, 2001.

[28] Liao L. Fox D. Patterson, D.J. and H. Kautz, “Inferring High-Level Behavior from Low-Level Sensors." Technical report, University of Washington, Seattle, WA, 2003.

[29] Laird J.E. Nielsen P.E. Coulter K.J. Kenny P.G. Jones, R.M. and F. Koss, “Automated Intelligent Pilots for Combat Flight Simulation”, AI Magazine, Vol. 20, No.1, pp.27-41, 1999.

[30] R. Jones and J. Laird, “Constraints on the Design of a High-Level Model of Cognition”, Technical report, University of Michigan, Ann Arbor, MI, 1996.

[31] B.S. Stensrud and A.J. Gonzalez, “Context-Based Reasoning: A Revised Specification”, Florida Artificial Intelligence Research Society (FLAIRS) Conference, 2004.

[32] M. van Lent and J.E. Laird, “Learning Hierarchical Performance Knowledge by Observation." In International Conference on Machine Learning, 1999.

[33] The RoboCup Soccer Simulator,” http://sserver.sourceforge.net/.

[34] Noda and H. Matsubara, “Soccer Server and Researches on Multi Agent Systems”, In H. Kitano, Proc. of IROS-96 Workshop on RoboCup, pp. 1-7, 1996.

[35] Carbonell J.G. Michalski, R.S. and T.M. Mitchell. Machine Learning: An Artificial Intelligence Approach, volume II. Morgan Kaufmann Publishers, Los Altos, CA, 1986.

[36] M.X. Zhou and S. Ma, “Toward Applying Machine Learning to Design Rule Acquisition for Automated Graphics Generation." 2000.

[37] H.K. Fernlund and A.J. Gonzalez, “An Approach towards Building Human Behavior Models Automatically by Observation." 2001.

[38] A.J. Gonzalez and R.H. Ahlers, “Context-Based Representation of Intelligent Behavior in Simulated Opponents”, In Computer Generated Forces and Behavior Representation Conference, 1996.

[39] B.S. Stensrud and A.J. Gonzalez, “Context-Based Reasoning: A Revised Specification”, Florida Artificial Intelligence Research Society (FLAIRS) Conference, 2004.

[40] T. Stanard and R. Hutton, “A Computational Model of Driver Decision Making at an Intersection Controlled by a Trafic Light." Technical report, Micro analysis and Design, Boulder, CO, 2001.

[41] G. A. Carpenter and S. Grossberg, “A Self-Organizing Neural Network for Supervised Learning, Recognition, and Prediction.” IEEE Communications Magazine, pp.UI63-6804, 1992.

[42] Grossberg S. Markuzon N. Reynolds J.H. Carpenter, G.A. and D.B. Rosen, “Fuzzy ARTMAP: A Neural Network Architecture for Incremental Supervised Learning of Analog Multidimensional Maps." In IEEE Transactions on Neural Networks, Vol. 3, 1992.

[43] G.A. Carpenter, S. Grossberg, and J.H. Reynolds, "ARTMAP: Supervised real time learning and classification of nonstationary data by. A self-organizing neural network”, Neural Networks, Vol. 4, pp. 565 - 588, 1991.

[44] W. Pedrycz and F. Gomide, “Fuzzy Systems Engineering: Toward Human-Centric Computing”, IEEE press, USA, 2007.

[45] H. Akiyama, D. Katagami, and K. Nitta, “Training of Agent Positioning Using Human’s Instruction”, Journal of Advanced Computational Intelligence and Intelligent Informatics, Vol.11 No.8, 2007.