15th Triennial World Congress, Barcelona, Spain

MULTI-AGENT SYSTEM BASED FUZZY CONTROLLER DESIGN WITH GENETIC TUNING FOR A SERVICE MOBILE MANIPULATOR ROBOT IN THE HAND-OVER TASK

Mustafa Suphi ERDEN[1,2], Kemal LEBLEBİCİOĞLU[1,2] and Uğur HALICI[1,2]

[1]Department of Electrical & Electronics Engineering, [2]Machine Vision and Intelligent Systems Laboratory.
Middle East Technical University.
06531, Ankara, Turkey.
suphi@metu.edu.tr, kleb@metu.edu.tr, halici@metu.edu.tr

Abstract: This paper presents an application of the multi-agent system approach to a service mobile manipulator robot that interacts with a human during an object delivery and hand-over task in two dimensions. The base, elbow and shoulder of the robot are identified as three different agents, and are controlled using fuzzy control. The fuzzy rules of each agent are written considering the state of other agents besides its own state. While writing the rules effective delivery and avoiding the contact with human not to cause physical harm is considered. The membership functions of the fuzzy controllers are tuned using genetic algorithms. In tuning, the performance is calculated considering the deviation from the optimum path, time spent to reach the human hand and energy consumed by the actuators. The proposed multi agent system structure based on fuzzy control for the object delivery task succeeded in both effective and safe delivery. *Copyright © 2002 IFAC*

Keywords: Agents, distributed control, fuzzy control, genetic algorithms, membership functions, delivery systems, robot arm.

1. INTRODUCTION

The multi-agent systems approach to distributed control systems is widely spread for important utilities it provides. They offer a decentralized control model based on agents. In this approach several elements, namely agents, cooperate with each other to reach their own goals, and these goals sum up to the final goal of the whole system. Therefore, the tasks to be accomplished to achieve a global goal are distributed between agents and each agent is to perform its special task. Each agent in the system has its own control block.

Fuzzy-logic controllers are shown to be very successful in complex systems where imprecision exists. With this method human decision making process is modelled with a collection of rules, and by these rules the linguistic terms are converted to precise numeric values. Once a collection of rules is assembled, it may be possible to tune the rule base for several specifications. Genetic algorithm is a powerful tool for optimisation in complex spaces, which is capable of jumping over the local extremes and approaching to the global extreme. It is a common usage to design the control mechanisms of agents in multi-agent systems with fuzzy controllers. And genetic algorithm is widely applied to fuzzy control rule bases to tune the membership functions of the rules. In this study it is aimed to apply tuning of the membership functions of the fuzzy controllers of each agent in a multi agent control system, using genetic algorithms.

The model used in this application is taken from Agah, *et al.* (1999): a service mobile manipulator robot that interacts with a human during an object delivery and hand-over task. In this task is the robot desired to accomplish the delivery in an *efficient* and *safe* way. Hence, the delivery should be performed with the minimum distanced path, in minimum time and dissipating minimum energy. Moreover, the contact with the human should be severely avoided. The agents defined in this system are robot base, robot shoulder and robot elbow. Each agent is controlled with fuzzy controllers. The fuzzy rules of each agent are written considering the state of other agents besides its own state. The membership functions of the fuzzy controllers are tuned using genetic algorithms after the fuzzy controller design is complete. In tuning, the tip points of the membership functions are arranged. The fitness function, hence the performance is calculated considering the deviation from the optimum path, time spent to reach the human hand and energy consumed by the elbow and shoulder actuators. The proposed multi agent system structure based on fuzzy control for the object delivery task succeeded in both effective and safe delivery.

2. LITERATURE REVIEW

In (Valesco, *et al.*) a distributed control problem is taken into account through multi-agent systems perspective and a general structure is presented. A MIX multi-agent platform is suggested to define a layered model of the network that provides agents with a uniform view of the net. This approach is applied to the case of economic control of a fossil fuel fired power plant. Agah, *et al.* (1999), presents the concept of fuzzy logic control based on distributed multi-agent systems, but it proposes a new approach in which the software agents cooperate and contend to control the system. The main controller in the system evaluates the proposals of the contending agents according to their relevance, confidence and effect, and then selects the winner action that will be taken by the system. Wangerman, *et al.* (1998), deals with applying the multi-agent systems approach to air traffic control. The principled negotiation between intelligent agents is the main subject of this paper. The structure of each agent is based on the functions it performs. These functions are classified as declarative functions, procedural functions, and reflexive functions.

Papers (Gurocak, 1999; Herrera, *et al.*, 1998; Pham and Karaboga, 1999) deal with tuning fuzzy controllers using genetic algorithms. Gurocak (1999) proposes the method of tunes the rule base by shifting the peak locations of the fuzzy sets of the system both in the antecedent and consequent of the rules. Genetic algorithm optimisation is used to find the peak locations that best fit to a desired input-output relation set. Herrera, *et al.* (1998), gives a grouping of genetic fuzzy systems depending on the knowledge base components (data base and rule base) included in the genetic learning process. In the paper a genetic fuzzy system methodology based on learning of complete knowledge base, both rules and membership functions, is proposed. Pham, *et al.* (1999), proposes a new adaptive fuzzy logic control employing neural network and genetic algorithms techniques. The identification is performed using a neural network model and training it on-line using genetic algorithms. Then the fuzzy logic controller best able to control the identified model is designed. In (N-Nagy, *et al.*, 1987; Koren, 1985), the mathematical approach to robot arm structure is presented in detail; also drawings for the servomotor block diagrams for the junctions are given.

3.SYSTEM DESCRIPTION

In the system is a service mobile manipulator robot that interacts with a human during an object delivery and hand-over task. The aim is to control the robot in order to deliver the object in an efficient and safe way. The structure is given in Fig. I with the parameters utilized in the control. In this system the robot has access to the x-y position of the human hand with respect to its own hand position (dx-dy). These are the main inputs to the robot.

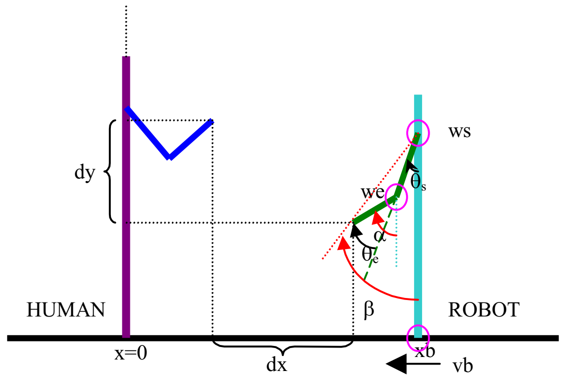


Fig. I: The human-robot system of object delivery task.

The distance of the robot base from the human base(xb), the shoulder angle(θ_s), and the elbow angle(θ_e) make up the state of the robot. The control parameters are robot base velocity(vb), shoulder angular velocity(ws), and elbow angular velocity(we). The control system is to modify these control parameters in an appropriate manner according to the input and state parameters of the robot. In this study it is assumed that the direct manipulation of the linear and angular velocities is possible, which is not the case in real-world applications where only the electrical inputs to the actuator motors are manipulated. However, the study will not suffer from this, since almost direct manipulation of motor speeds in small speed ranges is possible with the improved step-motor technology.

The human puts his hand forward to take the object and may change the position of his hand continuously. The robot has to track the hand and arrange its action continuously.

4. IDENTIFICATION OF AGENTS

The agents are identified according to the different tasks to be performed in the system. These are base agent, shoulder agent, and elbow agent (put in circles in Fig. I) for base manipulation, shoulder angle manipulation, and elbow angle manipulation respectively. The inputs and outputs of the agents are as in Table 1.

Table 1: Input and output parameters of the agents.

AGENTS	INPUTS	OUTPUTS
Base	dx, dy	vb
Shoulder	dx, dy, β	ws
Elbow	dx, dy, α	we

Base agent manipulates the base velocity of the robot according to the x-y distances between robot hand and human hand (dx-dy). The effect of the states of the shoulder and elbow agents with its own state, namely θ_s, θ_e and xb, are reflected to the base agent in dx and dy information, and this is enough for the task of base agent. When the elbow and base angles are held constant, the change in base position will directly affect the change in dx.

Shoulder agent manipulates the angular velocity of the shoulder according to dx-dy distances and β angle. The dx and dy information are not enough in this case since the state information of shoulder and elbow agents are not reflected to these for shoulder angle manipulation. However, the β angle reflects these to shoulder angle manipulation. When the base position and elbow angle are fixed, the shoulder angle manipulation will change the β angle directly, and this will be the contribution of the shoulder agent to whole systems action. Here β is equal to $\theta_s+\theta_e$, that it is the cooperated state information of states of elbow and shoulder.

Elbow agent manipulates the angular velocity of the elbow according to dx-dy distances and α angle. The α angle reflects the shoulder and elbow angle information for the elbow angle manipulation. When the base position and shoulder angle are fixed, the change in elbow angle will directly change the α angle, and only this will take effect in dx-dy manipulation.

5. RULES AND MEMBERSHIP FUNCTIONS FOR THE FUZZY CONTROLLERS OF AGENTS

The crucial part of this study is to construct the fuzzy rules of fuzzy controller of each agent. These rules will relate the outputs of the agent to the inputs. The following describes the logic in writing the rules for each agent.

Base Agent:

- Approach to human with high speed when dx is large.
- Approach slowly when dx is small but $|dy|$ is large.
- Approach very slowly when both dx and $|dy|$ are small.
- Go backwards very fast when dx is negative.

Shoulder Agent:

- When β is small (0^0-45^0) the change in θ_s will change dx much, but will change dy little.
- When β is big (45^0-120^0) the change in θ_s will change dx little but dy much.
- Make dx variations fast when dx is large.
- Make dy variations fast when $|dy|$ is large.
- Make the movement of shoulder very slow when the hands are very near, dx and $|dy|$ are very small.
- When dx is negative, don't approach to the hand, but escape from the hand making the shoulder movement in reverse direction according to the sign of dy, with a velocity proportional to $|dy|$. When the robot arm is below the human arm dy is positive, it is negative when the robot arm is above the human arm.

Elbow Agent:

- When α is small (0^0-45^0) or big (135^0-180^0) the change in θ_e will change dx much, but will change dy little.
- When α is normal (45^0-135^0) the change in θ_e will change dx little, but will change dy much.
- Make dx variations fast when dx is large.
- Make dy variations fast when $|dy|$ is large.
- Make the movement of elbow very slow when the hands are very near, dx and $|dy|$ are very small.
- When dx is negative, don't approach to the hand, but escape from the hand making the elbow movement in reverse direction according to the sign of dy, with a velocity proportional to $|dy|$. When the robot arm is below the human arm dy is positive, it is negative when the robot arm is above the human arm.

The membership functions for the input-output variables are defined as shown if Fig. II, III, and IV. Fuzzy rules are given in Tables 2, 3, and 4. These membership functions are not the tuned ones. They show the base structures on which the tuning is applied. Since there are three inputs for the shoulder and elbow agents, their rules need three tables.

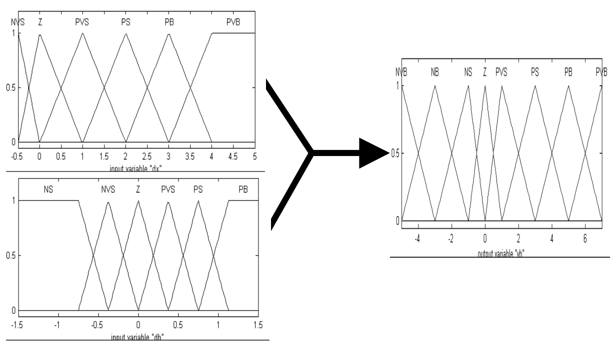


Fig. II. Input and output membership functions for the base agent.

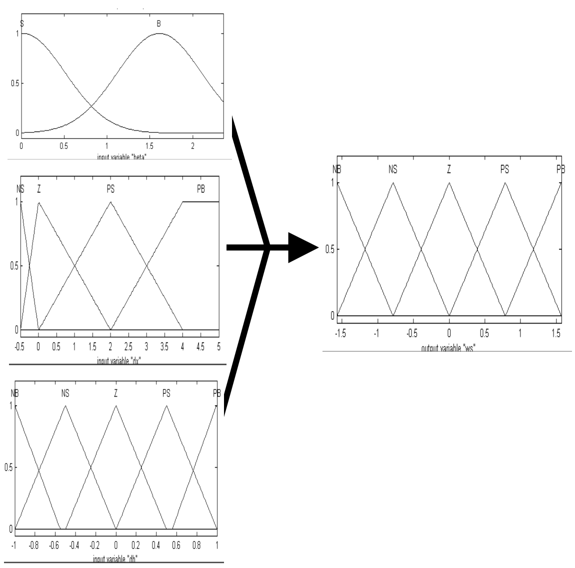


Fig. III. Input and output membership functions for the shoulder agent.

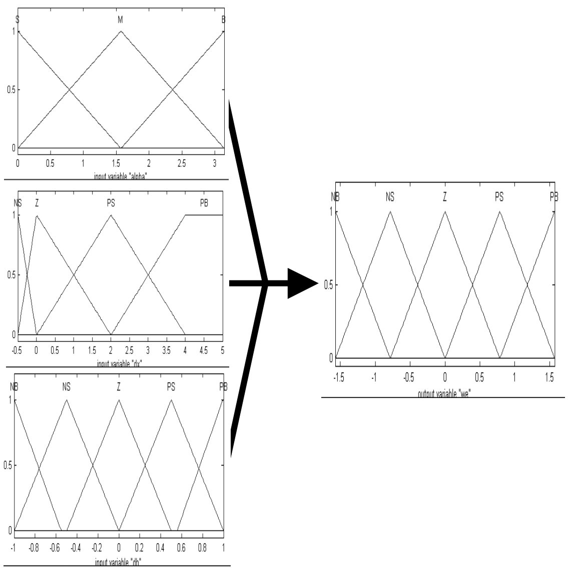


Fig. IV. Input and output membership functions for the elbow agent.

Table 2. Rule table for the base agent

dh dx	NS	NVS	Z	PVS	PS
NVS	NVB	NVB	NVB	NVB	NVB
Z	NS	Z	Z	Z	NS
PVS	PVS	PVS	PVS	PVS	PVS
PS	PS	PVS	PVS	PVS	PVS
PB	PB	PS	PS	PS	PB
PVB	PVB	PVB	PVB	PVB	PVB

Table 3. Rule table for the shoulder agent.

β=S (small)

dh dx	NB	NS	Z	PS	PB
NS	PB	PS	Z	NB	NB
Z	NS	NS	Z	PS	PS
PS	NB	NS	Z	PS	PB
PB	NB	NS	Z	PS	PB

β=B (big)

dh dx	NB	NS	Z	PS	PB
NS	PB	PB	NB	NB	NB
Z	Z	Z	Z	Z	Z
PS	NS	NS	PS	PS	PS
PB	NB	NS	PB	PB	PB

Table 4: Rule table for the elbow agent.

α=S (small)

dh dx	NB	NS	Z	PS	PB
NS	PB	PS	PS	NS	NB
Z	NS	Z	Z	Z	PS
PS	NS	NS	PS	PS	PS
PB	NS	NS	PB	PB	PB

α=M (medium)

dh dx	NB	NS	Z	PS	PB
NS	PB	PB	Z	NB	NB
Z	NS	NS	Z	PS	PS
PS	NB	NS	Z	PS	PB
PB	NB	NS	Z	PS	PB

α=B (big)

dh dx	NB	NS	Z	PS	PB
NS	PB	PS	NS	NS	NB
Z	NS	Z	Z	Z	PS
PS	NS	Z	PS	PS	PS
PB	NS	NS	PB	PB	PB

6. TUNING WITH GENETIC ALGORITHMS

Tuning is done by shifting the tip points of the membership functions in the range defined by the base (Fig. V). It is applied only on the input membership functions. Since the output membership functions of the controllers of all agents are dense enough they are considered not to need tuning. Using genetic algorithms necessitates defining the chromosome structure and fitness function for this application. The idea for chromosome structure is taken from Gurocak (1999). In this approach the chromosomes encode the place of the tip point in its range with 0's and 1's. 0000 assigns the leftmost of the range, 1111 assigns the rightmost of the range, and remaining four-combinations of 1's and 0's encode the regions in between. The whole chromosome is made up of all the codes for all membership functions (Fig. VI).

As mentioned before the fitness function is constructed by considering the deviation from the optimum path, time spent to reach the human hand, and energy consumed by the shoulder and elbow actuators. The deviation is calculated using the area between the actual path of the robot hand, and the optimum line between the initial positions of the robot hand and human hand (Fig. VII).

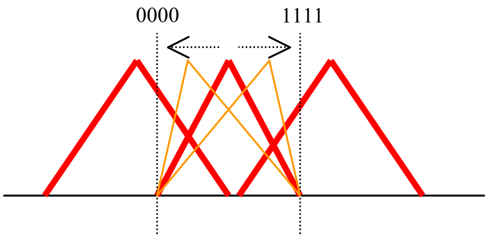


Fig. V. Coding the shfting of the tips of membership functions.

$\underbrace{1\,0\,0\,0}$ 1 1 1 0 1 0 1 0 0 1 0 1 1 1 0 1 1 0 0 1 $\underbrace{1\,0\,0\,1}$

For the first membership function. *For the last membership function.*

Fig. VI. Chromosome structure.

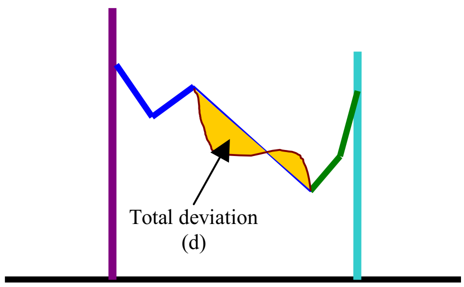


Fig. VII. Deviation of the route.

The energy calculations for the shoulder and elbow junctions are done using the formula $E = \int_0^t T(t).w(t).dt$, where T stands for torque and w stands for angular speed. The overall fitness calculation is done weighting the three factors. If the time spent is t_f, then the fitness is,

$$Fitness = C - (w_d.d + w_e.E + w_t.t_f) \qquad (1)$$

where C is a constant, and w_d, w_e, w_t are the weights for distance, energy, and time respectively.

For calculating the fitness of each chromosome the controller encoded by the chromosome is applied to the system. The sysytem is simulated beginning from the same robot hand position for all chromosomes while the human hand is stationary at the same point. The steps of the genetic algorithm for this application can be given as in Table 5. In applying the genetic algorithm roulette wheel selection, crossover and mutation operations are used for reproduction. The crossover and mutation probabilities are taken as 0.3 and 0.05 respectively. During the application of genetic algorithms the best five chromosomes obtained from the beginning are stored and updated after each population generation.

Table 5 Genetic algorithm steps for tuning.

Steps of the Genetic Algorithm

1. Initialise a population of chromosomes with random 0's and 1's.
2. Assign the fitness value of each chromosome in the population.
   - Decode the chromosome.
   - Construct the controller.
   - Simulate the system with this controller for a fixed initial robot hand position and stationary human hand position.
   - Calculate the fitness value as in (1).
   - Assign this value to the chromosome as its fitness.
3. Put the best five chromosomes aside and update a store hold for the best five chromosomes.
4. Construct the new population from the old one using roulette wheel selection (the probability of being selected is proportional to the chromosome's fitness value).
5. Apply crossover and mutation operations on the population.
6. If the predefined number of generation is not performed go to step 2.
7. Take the best chromosome from the best five store. (The best five store gives the best five chromosomes achieved during the whole process.)

7. SIMULATION RESULTS

Some graphical results obtained with different weights are given in Fig. VIII, IX, X, and XI. These are the graphs showing the path followed by the robot hand, with the optimum line. All four trainings are done for the same initial robot arm and stationary human hand positions. The weights of the fitness function are changed in each. The time spent to reach the human hand is also indicated as *finaltime* on the graphs.

The graphs are consistent with the weights used for each. Figure VIII is an average graph, Figure IX has the path with minimum distance deviation since the area enclosed is the minimum, and Figure X has the minimum time spent to reach the human hand. The meaning of Figure XI is not realizable from the figure, but in simulation the arm of robot makes slow motions, as if distributing the motion equally between base, shoulder, and agent. When the simulations of these four results are examined, the one most resembling the human motion is the one in Figure XI. Hence, it is likely that human tries to minimize the energy in his/her motions.

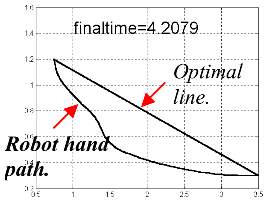


Fig. VIII. $w_d=1$, $w_e=1$, $w_t=1$

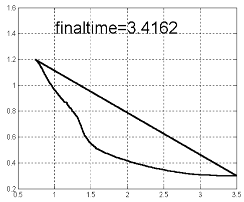


Fig. IX. $w_d=1$, $w_e=0$, $w_t=0$

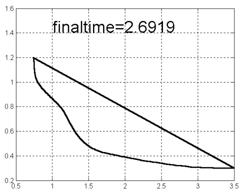


Fig. X. $w_d=0$, $w_e=0$, $w_t=1$

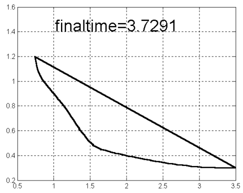
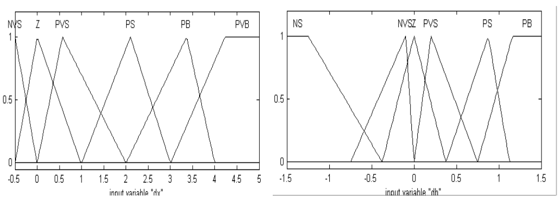


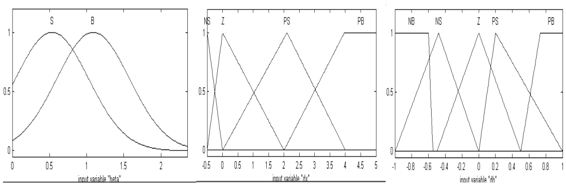
Fig. XI. $w_d=0$, $w_e=1$, $w_{t=}0$

Figures showing the paths followed by the robot hand, with the optimum lines.

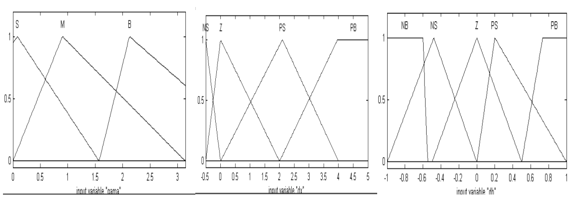
In Figure XII, the membership functions obtained after training (input membership functions of the agents) are given, for the case of Figure VIII.



Inputs of the base.



Inputs of the shoulder.



Inputs of the elbow.

Fig. XII. Tuned membership functions (inputs) of the fuzzy controllers of the agents for the case of Figure VIII.

8. CONCLUSION

In this study, multi-agent system approach to distributed controllers is applied to the system of a service mobile manipulator robot that interacts with a human during an object delivery and hand-over task in two dimensions. The agents are identified as the base, shoulder, and elbow of the robot. Each agent controls the movement of the junction referred by the name. Agents are controlled with fuzzy controllers. The rules for the fuzzy controllers are written considering both efficient and safe delivery. The multi agent system approach presented here seems to simplify the control considering both design and implementation. The results are satisfactory in both efficiency and safety. In this work, additional to the design, the training of the controllers is performed by tuning the membership functions with genetic algorithms. Optimum path, minimum time, and minimum energy considerations are made in the tuning process. Although the tuning improved the controller the improvement was not so significant. This is because the fuzzy rules written are robust with respect to the membership functions. A further study for this application may be to train the rules instead of the membership functions, or both at the same time. Another study may be performed to generalize the multi agent system based control for robots arms with more than two links that are moving in three-dimensional space.

REFERENCES

Agah, A. and K. Tanie (1999). Fuzzy logic controller design utilizing multiple contending software agents. *Fuzzy Sets and Systems*, **106**, 121-130.

Gürocak, H.B. (1999). A genetic-algorithm-based method for tuning fuzzy logic controllers. *Fuzzy Sets and Systems*, **108**, 39-47.

Herrera, F., M. Lozano and J.L. Verdegay (1998). A learning process for fuzzy control rules using genetic algorithms. *Fuzzy Sets and Systems*, **100**, 143-158.

Koren, Y. (1985). *Robotics for Engineers*, McGraw-Hill Inc.

N-Nagy, F. and A. Siegler (1987). *Engineering Foundations of Robotics*, Prentice-Hall International (UK) Ltd.

Pham, D.T. and D. Karaboga (1999). Self-tuning fuzzy controller design using genetic optimization and neural network modeling. *Artificial Intelligence in Engineering,* **13,** 119-130.

Valesco, J.R, J.C. Gonzalez, L. Magdalena and C.A. Iglesias. Multi-agent based control systems: A hybrid approach to distributed process control.

Wangermann, J.P. and R.F. Stengel (1998). Principled negotiation between agents: a model for air traffic management. *Artificial Intelligence in Engineering*, **12**, 177-187.