Analysis of an Enhanced Opportunistic System for Grasping Curved Objects through Rolling Contact

Andrea A. Ordean and Shahram Payandeh
Experimental Robotics Laboratory (ERL)
School of Engineering Science
Simon Fraser University
Burnaby, British Columbia, V5A 1S6, Canada

Abstract

This paper presents an analysis and design of an Enhanced Opportunistic System (EOS) and its implementation within the task of grasp planning through haptic exploration by rolling contact. The EOS is a modular system which has a great potential of encompassing a variety of grasp planning strategies under one controller and making them work together. This paper explores the mechanism which can make this possible. A novel rating system, embedded within the agents of the opportunistic architecture, is a part of the control mechanism of the EOS allowing the agents to estimate their own worth. The rating system consists of two sub-rating components; fixed and variable. The fixed sub-rating represents the predefined ranking of the agents while the variable sub-rating represents the opportunistic ranking of the agents. The performance of EOS is evaluated so as to determine the utility of the proposed two sub-rating system within the task of grasp planning. The evaluation consists of two parts: a) establishing what is gained by the presence of both the sub-ratings and b) to investigate the resultant grasps of the EOS.

1 Introduction

The analysis and design of an Enhanced Opportunistic System (EOS) is presented here within the context of grasp planning.

The EOS can bring all existing computational algorithms together and leave room for any future ones. The four properties of grasp planning sighted by Shimoga [10], grasp dexterity measures, grasp equilibrium, grasp stability, and grasp dynamic behaviour can all be combined under one roof and be made to work together within the EOS. However, the focus of this paper is the mechanism which would allow the four above-mentioned properties to work in a complementary fashion.

The EOS is an autonomous system which can deal with the complex problem of grasp planning. Like the Blackboard architecture [5], the EOS has some very redeeming properties such as: modularity and flexibility, within a centralized control.

Overgaard et al. [8] have introduced a multi-agent framework in grasp planning, which is very similar to the notion of the Blackboard. Five types of agents, also known as “Knowledge Sources” within the Blackboard framework work together to achieve a goal. Occello and Thomas [7] used a parallel structure based on the Blackboard architecture to control small mobile robots.

Halpern et al. [6] have acknowledged the need for agents of multi-agent systems to compute their own knowledge. The authors distinguish between two types of knowledge: externally ascribed knowledge and explicit knowledge. The former is the type of knowledge which the system programmer gives to the system, while the latter is the type of knowledge, which the system acquires through its inputs. The explicit knowledge is what allows the agents to act in a certain way. We are making use of these two types of knowledge, as Halpern et. al. have done, but we also use this knowledge to rate the utility of the agents of the EOS. The utility of an agent is its usefulness factor in the current situation.

Although the EOS is a knowledge-based system, it is not strictly primitive driven as the knowledge-based systems reviewed by Shimoga [10]. The system is set up so that it can take advantage of arising, unforeseen opportunities. This opportunistic nature is possible because of the enhancements made to the classic architecture through the addition of agent knowledge calculators. The agent knowledge calculators consist of a rating system based on the Bayesian Formalism.

In general, the easiest way for the controller to pick an agent is to give them each a rating and then select the highest rated agent. This rating could be preassigned such that the sequence of the agent execution is encoded within the assigned values.

The EOS rating method is based on a Bayesian formalism. The idea of augmenting the Blackboard architecture with the help of Bayes' Rule is not new. However, up to now, it has been applied only to evidence incorporation and hypothesis generation[4][11] in the area of image and speech recognition. Nonetheless, the idea can also be applied to rating agents and even to the way in which the controller chooses the most appropriate agent.

Haptic explorations of curved shapes have been investigated by Charlebois et al. [1], as well as by Chen et al. [2]. Both studies have looked at curvature estimation of objects through rolling contact, but have presented their results in slightly different ways. Charlebois et al. have identified two types of exploratory procedures(EPs). The first, requires one fingertip to roll, without sliding, and the second requires three fingers to be dragged across the surface of the object while keeping the object fixed.

This paper introduces a mechanism, consisting of a rating system, which allows agents to calculate their own knowledge in grasp planning and which also introduces an opportunistic component. A system, which incorporates this mechanism, and an analysis thereof is presented within the next few sections. Section 2 provides some background information for the rating system and for the exploratory procedure used. Section 3 will discuss the system architecture and indicate how the system was implemented. The experimental results are presented in section 4, while the analysis is left to section 5. Section 6 presents the conclusions and section 7 discusses possible future work.

2 Background Information

The rating system and the exploratory procedure used are going to be explained in more details in the next two subsections.

2.1 Rating System

The Bayesian Formalism makes it possible to reason in the presence of uncertainty and the manner in which probabilistic knowledge is dealt with can be used to draw a parallel between this probabilistic notion and the proposed rating system.

If we let $P(B_i)$ be the probability of sub-rating, B_i, occurring, then we can also look at $P(B_i)$ as a weighting of the sub-rating B_i. Also, if we let A be an agent, then $P(A,B_i)$ is the probability of both A and B_i occurring at the same time. One of the basic Bayesian axioms says that, given a set of n mutually exclusive variables, B_i, then the probability of the variable A, P(A), can be calculated from the probability of A and B_i, $P(A,B_i)$ [9],

$$P(A) = \sum_i (P(A, B_i)) \tag{1}$$

Then, using Bayes' Rule, $P(A,B_i)$ can be calculated by:

$$P(A|B_i) = \frac{P(A, B_i)}{P(B_i)} \qquad or \qquad P(A, B_i) = P(A|B_i) \times P(B_i) \tag{2}$$

where $P(A|B_i)$ is the probability that A will happen given B_i or $P(A|B_i)$ can be the rating of an agent A with respect to sub-rating B_i. Two requirements are imposed on Equation (2):

$$0 \le P(A) \le 1 \quad where\ A\ is\ any\ variable,\ or\ agent \qquad and \quad \sum_i P(B_i) = 1 \tag{3}$$

Combining equations (1) and (2),

$$P(A) = \sum_i (P(A|B_i) \times P(B_i)) \tag{4}$$

Continuing the parallel, P(A) in Equation (4) represents the total rating of agent A.

The agents could be given arbitrary ratings from the start, but this would not allow the opportunistic nature of the system. However, this type of assigned rating is not of total lack of use, because a pre-defined rating could provide a default for the system and there is a certain component of the rating which can be made to be opportunistic. As a result, part of the rating can be made to be influenced by the state of the system.

We have shown how Equation (4) can be used to combine several competing sub-ratings into one. Now, the next question that we must answer is "How many types of subratings are required?" Assuming that there are "m" agents and n sub-ratings,

$$P(A_j) = \sum (P(A_j|B_i) \times P(B_i)), \quad where\ \ 0 < j \le m \tag{5}$$

Therefore, for every agent, "n" multiplications are required. With "m" agents, a total of "mxn" multiplications are require each cycle for the rating calculation alone. Furthermore, if we have "n" agents (i.e. m = n) and "n" sub-ratings, then we need a total of "n^2" multiplications are required every cycle. Consequently, the lower "m" is, the less number of multiplications are required. However, we are not just talking about computing time. If we use the idea of agents rating each other, as in [4] and [11], then we end up with "n" agents and "n" sub-ratings. By interweaving all agents in this manner, it would make the addition or deletion of agents to the system hard to accomplish. Any such changes would call for a complete redesign of the ratings systems of every agent. As a result, much flexibility and all modularity is lost.

The main disadvantage of this is that the calculations can become very lengthy. In addition, this would cost the EOS its modularity and flexibility. By interweaving all agents in this manner, it would make changes to the system hard to accomplish. Any addition or deletion of agents would call for a complete redesign of the ratings system of every agent.

Thus, a more realistic number of sub-ratings must be defined. Here we are proposing a more realistic number of sub-ratings would be three. One of the sub-ratings could be a fixed, default value. The second can be a variable, opportunistic sub-rating which changes as a function of the state of the planner. Finally, the third could be a “Learned” rating. The “Learned” rating would be a parameter which gets updated constantly and can be specific type of object to be grasped or a particular orientation of the object to be grasped. In this paper only the first two types of sub-ratings will be discussed.

Two sub-ratings give more flexibility than one, yet the architecture is maintained relatively simple. For example, let the default sub-rating of agent j be $P(A_j|B_d)$, and the opportunistic sub-rating be $P(A_j|B_o)$. Then, the overall rating for each agent j can be $P(A_j)$, after combining the sub-ratings with respect to the pre-defined default weighting factor $P(B_d)$, and opportunistic weighting factor $P(B_o)$ as indicated in equation (6).

$$P(A_j) = P(A_j | B_d) \times P(B_d) + P(A_j | B_o) \times P(B_o) \qquad (6)$$

Note that while the sub-rating values are agent specific, the weighting factors are not. The following is an example of such a system:

Let's say that we have four agents, [Agent1, Agent2, Agent3, Agent4], and believe that we have equal confidence in their default sub-rating and in the opportunistic sub-rating, then the weight of the default rating and the weight of the opportunistic rating will each be 0.50. If the default, $P(A_j|B_d)$, and opportunistic, $P(A_j|B_o)$, sub-ratings of the agent A_j (where A_j is one agent and $1 \le j \le 4$), are as indicated in Table 1, then, using Equation (6) the total rating of Agent1 is calculated as follows:

$$\begin{aligned} & P(A_j | B_d) \times P(B_d) + P(A_j | B_o) \times P(B_o) \\ & = 0.90 \times 0.50 + 0.00 \times 0.50 \\ & = 0.45 \end{aligned} \qquad (7)$$

Consequently, the total rating of the agents is as shown in Table 1.

Table 1: Rating System ($P(B_d) = 0.5$; $P(B_o) = 0.5$)

A_j	$P(A_j\|B_d)$	$P(A_j\|B_o)$	$P(A_j)$
Agent1	*0.90*	*0.00*	*0.45*
Agent2	*0.60*	*0.80*	*0.70*
Agent3	*1.00*	*0.20*	*0.60*
Agent4	*0.40*	*0.50*	*0.45*

As a result, Agent2 has the highest rating among the four agents.

The default sub-rating, $P(A_j|B_d)$ is assigned to each agent at the time of creation of the agent. It is a fixed value throughout the program execution.

The opportunistic sub-rating, $P(A_j|B_d)$, will be re-set with each cycle. A cycle consists of the determination by the controller of the agent whose action is to be executed and the execution of that action. In any given cycle, the agent which executes its action may also change the sub-rating of any other agent, depending on the current status of the system. The sub-rating of an agent may also be modified at the time of each agents preconditions are tested. This allows the agents to take advantage of sensory information.

The opportunistic sub-rating need not be determined in the same way every time. One of the better ways of determining what the opportunistic rating should be is to make the rating a function of the system status. For example, when the robotic hand is about to grasp the object, after having executed one or more exploratory procedures, the system may choose to bring the hand closer in to the object (i.e. actuate the wrist), or to close the fingers to grasp the object (i.e. actuate the joint angles of the fingers). In this case, the choice is to be determined by the ratio of the estimated diameter of the object to the maximum vertical span that the robotic hand can achieve. If the object is relatively small, then it is necessary to bring the hand, closer to the object, so as to ensure that the object is not missed when the fingers are actuated. If the object is relatively large, then the fingers can be actuated so that they can grasp the object, without having to worry about missing the object. If the ratio is greater than one, then the object is determined to be too big to grasp. Consequently, the ratio is always less than 1. This ratio can then be used as the opportunistic sub-rating value for the joint angle actuation agent, and the complement of the ratio can be used as the value for the opportunistic sub-rating of the wrist agent. This is one example how the rating of an agent can be influenced by the current state of the system.

2.2 EP1

The haptic exploration investigated is that of a rolling finger on the surface of the object which was defined as EP1 by Charlebois et al. [1]. There is an EP2 which adds on to the shape estimation capabilities of EP1, however, its implementation was left for future work.

EP1 can be used to identify spherical or flat surfaces. It is executed by slightly rolling the robot finger in the neighbourhood of the contact point on the object. The rolling must be done at a known and constant angular velocity around a fixed axis in the instantaneous contact frame. This curvature estimation method is based on the following equation:

$$p = M^{-1} \times (K_1 + \tilde{K}_2)^{-1} \times \left(\begin{bmatrix} -\omega_x \\ \omega_y \end{bmatrix} - \tilde{K}_2 \times \begin{bmatrix} v_x \\ v_y \end{bmatrix} \right) \quad (8)$$

where,

p = contact point on probe in [u,v] direction
M = fingertip metric
K1 = curvature form of fingertip(known)
= curvature form of the object in contact with the fingertip
[ω_x, ω_y] are angular velocities of the fingertip's contact frame w.r.t. the object's contact frame around the x and y axes
[v_x, v_y] are the linear velocities of the fingertip's contact frame w.r.t. the object's contact frame in the x and y directions ([v_x, v_y, v_z] = [0,0,0] without slippage)

can be solved for and the diagonal elements of give the normal curvatures in the u and v directions.

$$\tilde{K}_2 = \begin{bmatrix} k_u & \\ & k_v \end{bmatrix} \quad (9)$$

The type of information which can be retrieved about an object with EP1 is the surface curvature, radius (r), of the object at a point on the object.

$$r_u = \frac{1}{k_u}, \quad and \quad r_v = \frac{1}{k_v} \quad (10)$$

This information can then be used to reconstruct the object and estimate grasping points on the object.

The complement of the relative error between the real and the estimated value of the radius can be used as an opportunistic rating for the agent which executes the EP1. Thus, if the error is small than EP1 is going to be executed less often than when the error is large.

The exploratory procedures are implemented as agents of the EOS and as a result they will be subject to the rating system added on to agents. For example, the sub-rating is influenced by the system being in the “info” state and by lack of knowledge about the object. As long as the object shape cannot be postulated with at least a 60% confidence, while in the “info” state, EP1 will be called on repeatedly.

3 System Architecture

Similarly to the Blackboard architecture, the EOS comprises mainly of three types of structures: the controller (C), the agents (A), and the information board (IF). Figure 1 shows the information/data flow among the different components which make up the EOS.

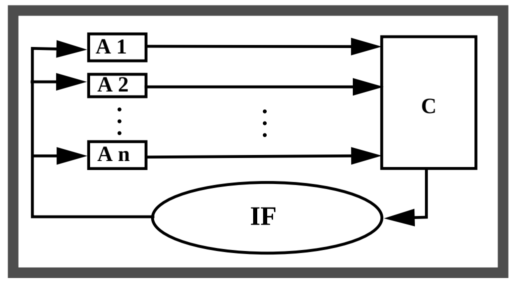

Figure 1 : The EOS Architecture

This system has been implemented mainly in PROLOG with the exception of the C subroutines which simulate the EP1 execution.

The EOS has three stages: search, information, and grasping. The first stage is concerned with locating the object in a three dimensional space of given dimensions. Once the object has been located, the information stage ensures that enough data is gathered about the object so that the system has a good idea of what the object is. Once the object has been (partially) identified, then the grasping stage deals with going about and getting a grasp on the object.

The agents each take the current situation under consideration and independently suggest a possible course of action. Each agent is implemented as a module within PROLOG.

Currently, the system consists of eight agents: end, finger1, finger2, finger3, wrist, ep1, post_shape, grasp_-points. The agents are restricted to participate only during certain stages. The reason for this is to reduce the number of useless “suggestions” made by the agents. For example, the ep1 agent will only participate during the stage of information gathering, so there is no reason to waste resources calculating a rating for this agent during the search or grasping stages. Similarly with the post_shape agent, which postulates the shape of the object given the gathered information.

The controller’s decision is a function of the total rating of each agent in a cycle. Given that each agent can produce a rating for itself, currently, the controller simply chooses the agent with the highest rating.

4 Experimental Results

As mentioned in Section 3, the system runs according to three stages. Consequently, a typical scenario involves the robotic hand starting off at a given location in space and trying to locate the object. This is a random search in which the direction [+x, -x, +y, -y, +z, -z] is picked randomly, and so is the step size taken in the chosen direction. Once the object is located, the system will try to get the robotic hand to identify the object by executing the EP1 on the object. If the confidence in the results of the procedure are high, then the robotic hand will attempt to grasp the object. The type of grasp used in this system is the precision grip [3]. The precision grip is a type of grasp in which only the fingertip of each finger is in contact with the object. Both sub-rating weighting factors were 0.50.

The analysis of the EOS was performed in two parts: a) Compare the absence vs. the presence of the variable sub-rating component, so as to establish whether the conjunction of the two sub-ratings is valid; b) Investigate the resultant grasps of the EOS.

4.1 Default vs. Opportunistic Sub-rating

We assigned a 1.0 weighting factor to the default sub-rating and a 0.0 weighting factor to the opportunistic sub-rating and the EOS executed each of the [finger1, finger2, finger3, wrist] agents once and stopped. These agents were all allowed to contribute during the search stage.

Next, a 0.0 weighting factor was assigned to the default sub-rating and a 1.0 weighting factor was assigned to the opportunistic sub-rating. The EOS made it through the search and information stages correctly, but stopped dead when it reached the grasp state.

4.2 The EOS Resultant Grasps

The grasps which this systems aims for is a precision grip. In addition, since only spheres where tested in this preliminary stage, the goal was to achieve a “good” grasp where the fingers make contact with the sphere about its diameter. Figure 2 shows what is referred to by the term “good” grasp. However, as long as the fingertips of the three fingers made contact with the sphere, the grasp result was included in Figure 3. If the described grasp was not achieved, then the data point was not included in the grasp.

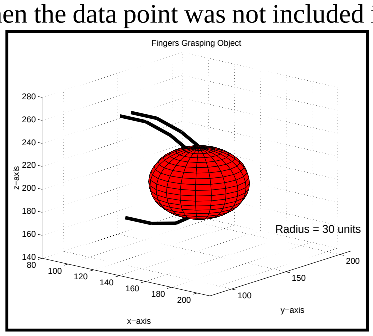

Figure 2 : “A Good Grasp”

The robotic hand consists of three fingers and each finger has three links. All links have the same length. The number of grasp cycles were noted as a function of the ratio of the diameter of the object to the height of the robot hand. The height of the robot hand is defined as the vertical distance between the upper two fingers and the bottom finger. The reason for using the ratio was to be able to compare the data on an even basis. Figure 3 shows a summary of that data.

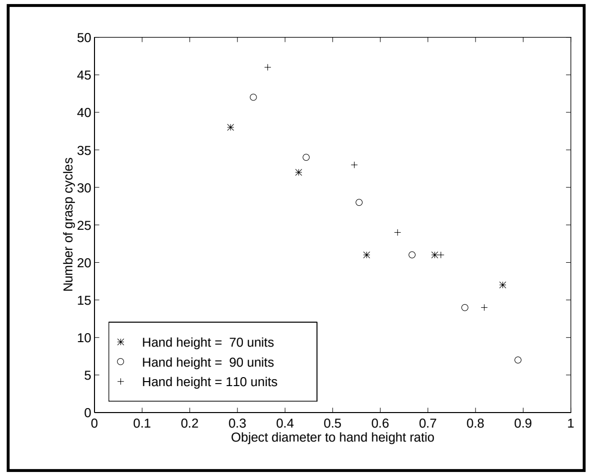

Figure 3 : Object Diameter to Hand Height Ratio vs. Grasp Cycles

5 Analysis

As seen in section 4.1, neither the default nor the opportunistic sub-rating function well independently. The default sub-rating simply pre-defines the order of executing agents. Consequently, this sub-rating alone provides no flexibility.

On the other hand, even the flexible, opportunistic sub-rating was not self-sufficient. The fact that the system got through the search and information stage is purely coincidental. At the beginning, of a new stage all opportunistic sub-ratings are zero. Consequently, the controller will select the first agent in the list. If this agent is an appropriate one to select at the time, it may help invoke the opportunistic nature. However, if the agent selected does not know what to do in the given situation, than the program will terminate.

At the beginning of the search stage, the “finger1” agent was chosen. This agent happened to know what to do, the search stage proceeded as usual. The information stage was lucky as well, because the “ep1” agent was executed by virtue of being the first in the list, which then provided some information for the “post_shape” agent to ponder on. However, the same luck did not occur during the grasping stage. “finger1” agent was picked again to start off and this time the agent did not know where to begin, as it did not have enough information.

Consequently, we have seen that only one of the sub-ratings is not enough to have a working, opportunistic system. As a result of this experiment, we have discovered an emergent role for each of the sub-ratings. The opportunistic sub-rating is the driving force which gives the system its flexibility, however, the default sub-rating is required to give the system its stability.

The experiment results shown in section 4.2, indicate that as the ratio of the diameter of the object to the height of the hand increases, the number of cycles required to complete a grasp decreases. This result can be explained by the method in which the grasping state was implemented. Once the system believes that it has enough information about the object, it tries to grasp the object. At this time it must choose among four agents, [finger1, finger2, finger3, wrist]. If the finger agents are selected, then the joints of the corresponding finger are actuated by a small amount. If the wrist gets chosen, then the robot hand will move a little closer to the object. The reason for this is that for small objects, the wrist has to be closer to the object so that the object is not missed as the joints are actuated. Consequently, if an object is relatively large, the fingers need only be actuated a few times until the precision grasp is achieved. However, for small objects, the finger and wrist agents take turns, making their way to the object slowly, but surely.

6 Conclusions

This paper has introduced a novel rating system for the agents of a multiagent modular system. The multiagent system designed is an inclusive type of system, which encompasses all grasp planning strategies. The rating scheme is based on both default and opportunistic knowledge. As the experiments have shown, the use of both types of knowledge vs. the use of either of the types of knowledge alone is the difference between a system which works vs. one that does not. A system that works can complete all three stages of the task and exit gracefully whether the resultant grasp has been successful or not. We have also shown, that the ratio of the diameter of object to the hand height is inversely proportional to the number of grasping cycles required to complete the grasp. This result is due to the manner in which the opportunistic sub-rating is evaluated.

7 Future Work

The addition of another exploratory procedure will be pursued. EP2 is executed by dragging three fingers across the surface of the object. By doing so over a large surface, the surface can be approximated with a small number of sampled points.

The third sub-rating, which was left out in this paper, can be added on to the rating system. This sub-rating represents the “learning” effect. For example, if the system detects that the current object being explored is similar to another object which was previously explored, then the system can use the learned sub-rating to influence the coarse of action so that the grasp which was previously used can be tried out on the current object, thus achieving the grasp in a fewer number of cycles.

Currently, the controller selects the agent with the highest rating,. A future possibility is for the controller to select the agent in the presence of higher goals or other higher level information, Info1 and Info2. In this case, the controller would need to have a means for further screening the agents, A, B, and C, using the ratings of the agents as inputs to its screening mechanism.

The screening mechanism alluded to in the previous paragraph can also be devised from the ideas of a Bayesian Network (BN). In this case, an actual layout of the BN may need to be drawn to establish the relationship between inputs, [A, B, C, Info1, Info2], and outputs, [X, Y, Z]. Figure 4 illustrates a simple two-stage BN:

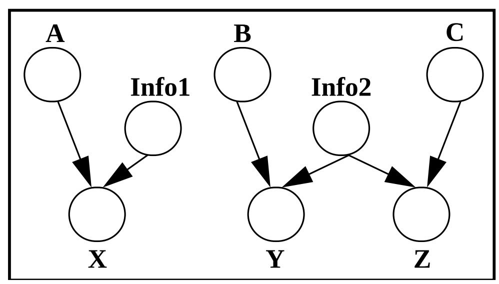

Figure 4 : Controller Screening Mechanism

Thus the controller would now pick the action which maximizes one of the outputs, X, Y, or Z.

Another possible direction for future work may be to look into the utility of giving the controller the ability to modify the weights of the types of sub-ratings, either on a per agent basis or on the whole.

References

[1] M. Charlebois, K. Gupta, and S. Payandeh “Curvature Based Shape Estimation Using Tactile Sensing,” IEEE International Conference on Robotics and Automation, 1996, pp. 3502 - 3507.

[2] N. Chen, R. Rink, and H. Zhang “Local Object Shape From Tactile Sensing,” IEEE International Conference on Robotics and Automation, 1996, pp. 3496 - 3501.

[3] M.R. Cutkosky, Robotic Grasping and Fine Manipulation, Kluwer Academic Publishers, 1985, p. 130.

[4] L.D. Erman, F. Hayes-Roth, V.R. Lesser, and D.R.

Reddy “The Hearsay-II Speech-Understanding System: Integrating Knowledge to Resolve Uncertainty” in Engelmore and Morgan (editors), Blackboard Systems, Addison-Wesley Publishing Company, 1988, pp. 61 - 64.

[5] B. Hayes-Roth “The Blackboard Architecture: A General Framework for Problem Solving?,” Technical Report HPP-83-30, Stanford University, 1983.

[6] J.Y. Halpern, Y. Moses, and M.Y. Vardi “Algorithmic Knowledge,” Proceedings of the Fifth Conference on Theoretical Aspects of Reasoning About Knowledge (TARK), 1994, pp. 255 - 266.

[7] M. Occello and M.C. Thomas “Systèmes Multi-Agents Temps Réel: Un Modèle d’Organisation basé sur le Concept de Blackboard”, Revue d’Intelligence Artificielle, pp. 1 - 25

[8] L. Overgaard, B.J. Nelson, and P.K. Khosla, “A Multi-Agent Framework For Grasping Using Visual Servoing and Collision Avoidance,” IEEE International Conference on Robotics and Automation, 1996, pp. 2456 - 2461.

[9] J. Pearl. Probabilistic Reasoning in Intelligent Systems, Morgan Kaufmann Publishers, 1988.

[10] K.B. Shimoga “Robot Grasp Synthesis Algorithms: A Survey,” The International Journal of Robotics Research, vol. 15, no. 3, June, 1996, pp. 230 - 266.

[11] A. Tailor “MXA - A Blackboard Expert System Shell” in Engelmore and Morgan (editors), Blackboard Systems, Addison-Wesley Publishing Company, 1988, pp. 315 - 332.