LEARNING VERSUS ANALYTICAL APPROACH TO CONTACT ESTIMATION IN ASSEMBLY TASKS WITH ROBOTS *

R. Suárez†, L. Basañez†, M. Nuttin‡, H. Van Brussel‡ and J. Rosell†.

† Instituto de Cibernética (UPC-CSIC), Diagonal 647, 08028 Barcelona, SPAIN (e-mail: suarez@ic.upc.es).
‡ K.U.Leuven-PMA, Celestijnenlaan 300 B, B-3001 Heverlee, BELGIUM (e-mail: nuttin@mech.kuleuven.ac.be).

Abstract - *In order to automatically execute assembly tasks with robots following a fine-motion plan, it is usually necessary to estimate the current contact situation to select the corresponding robot command. This contact estimation is generally based on configuration and force sensory data. Two methods for this purpose are presented here: a) an analytical method which explicitly takes into account all the uncertainty sources that may affect the task, and which computes the sets of configuration and forces compatible with each possible contact situation; and b) an inductive learning approach based on a backpropagation neural net that uses simulated contact-situation examples for the training phase. The methods are illustrated by the simple assembly task of positioning a block into a corner considering three degrees of freedom. The advantages and disadvantages of each approach are also discussed.*

Key Words - *Robotic Assembly, Fine-motion Planning, Contact Estimation, Learning, Neural Nets.*

1. INTRODUCTION

One promising approach to the automatic programming of assembly tasks with robots is the automatic generation of assembly plans, which usually contain the actions to be executed from each task state in order to achieve the assembly goal. These states can be defined in different ways (although sometimes it is not explicitly done), such as: using fuzzy set theory [14] [1], performing partitions of the configuration space [13] [5], or classifying the possible contact situation between the objects to be assembled [6] [18].

In any case, two main problems arise when uncertainty is considered: a) the identification of the current task state, and b) the automatic determination of the proper movement (i.e. the robot command for the controller) for each task state.

Different solutions have been proposed in order to solve these problems, giving rise to planners that explicitly identify the task state to decide which command must be applied [13] [5] [6] [8] [12] [18] on one hand and, on the other, to reactive systems that directly map sensor information into a robot command [16] [10] [9] [14]. This paper follows the first approach dealing with the problem of current task state estimation in order to follow the assembly strategy determined by a fine-motion planner.

* This work was partially supported by the ESPRIT Project B-LEARN II under contract n°7274, the CICYT project TAP93-0415, and the Interuniversity Attraction Pole IUAP-50, initiated by the Belgian Ministry of Science Policy. J. Rosell has a grant from the Catalan Government.

It is assumed here that task states are defined by the basic contacts between the objects to be assembled, i.e. each task state represents a different contact situation between the objects [18]. Information from two types of sensors is used for on-line estimation: configuration (the relative position and orientation between objects) obtained from knowledge of the environment and the robot configuration, and generalized reaction force (reaction force and torque) obtained from a force/torque sensor in the robot wrist.

The three degrees of freedom task of positioning a block into a corner (figure 1) is used to illustrate the method. It is assumed that the block can be positioned closely enough to the corner by gross motion; then, nine different contact situations are possible according to the nominal model of the objects (figure 2). Actual configurations in which each contact situation can be reached depend on the deviations of geometric variables such as object shape and size, robot positioning, and undesired slippings of the object in the robot gripper. The effect of these sources of geometric uncertainty on the contact configurations has been modeled in [2], and their effect on the possible reaction forces in [17]. Making use of these uncertainty models, a simulator capable of dealing with random deviations of the parameters has been implemented.

Two different estimation approaches are presented and discussed in this paper: an analytical one, based on geometric reasoning over the models of the objects and the different sources of uncertainty, and a learning approach, based on data obtained

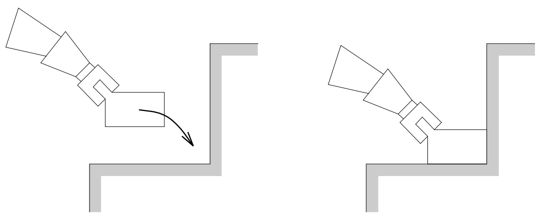

Fig. 1. *Positioning of a block into a corner.*

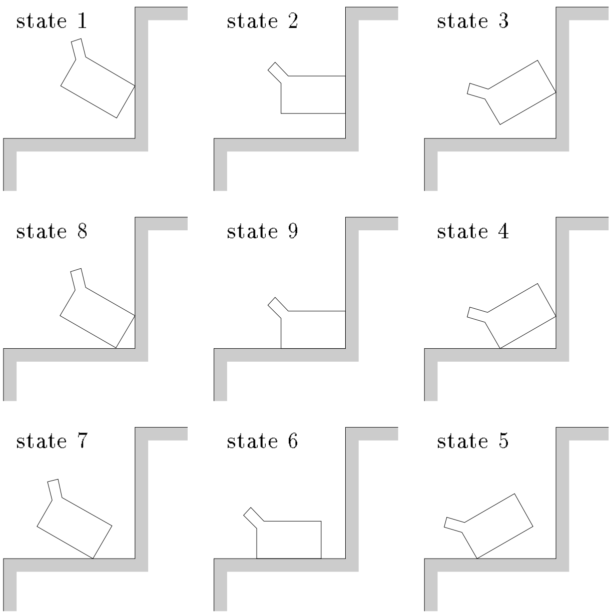

Fig. 2. *Nine possible nominal states.*

from real task executions performed by a teacher, or from simulations. The paper is organized as follows: section 2 and 3 respectively describe the analytical and learning approaches, in section 4 a discussion about the advantages and disadvantages of each approach is presented and, finally, section 5 summarizes the conclusions of the work.

2. ANALYTICAL APPROACH

2.1. Methodology

The models of the geometric uncertainty sources allow to analytically determine, for each task state, the set of configurations in which the state may occur, called *configuration realization domain*, DCr, and the set of possible generalized reaction forces that can appear in these configurations, called *force realization domain*, DGr. Then, by adding the uncertainty of the corresponding sensors, it is possible to obtain the sets of configurations and reaction forces that can be sensed when the state takes place; these are called *configuration observation domain*, DC, and *force observation domain*, DG, respectively.

The analytical method of contact estimation is based on the domains DC and DG, and can be divided into the following two phases:

- **Off-line** determination of the domains DC and DG for each possible contact situation between the objects to be assembled. In this phase, *geometric uncertainties*, *sensor uncertainties*, and *friction* are considered.
- **On-line** matching of the sensed configuration and force with the off-line computed domains. The sensed configuration is initially considered, and if it matches with more than one domain DC, then the sensed force is used; nevertheless, due to uncertainty it is possible that more than one contact situation be compatible with the sensed data.

2.2. Off-line computation

Configuration Observation Domains

Configuration domains are built by merging the different geometric uncertainties in the configuration space [2]. First, the uncertainty regions in the physical space where the vertices and edges of the objects can lie are determined. Figure 3a shows these regions for the block and the corner vertices and edges.

Once the condition for the existence of a basic contact has been established, i.e. a vertex of one object against an edge of the other, the corresponding regions of uncertainty in the physical space are mapped into the configuration space; this gives rise to the uncertainty domain DCr of a state with only one basic contact. So, the domain DCr includes the C-face containing the corresponding real contact configurations. The domain DC is obtained by adding to DCr the uncertainty of the configuration sensor (the robot itself). Figure 3b shows the domains DC of the task states with only one basic contact, and figure 3c is a section of these domains for the orientation of the block shown in figure 3a, the center of the small circle indicating the configuration (i.e. position) of the block.

The domains DCr and DC of the states with more than one basic contact are obtained as the intersection of the corresponding domains of the involved basic contacts. Nevertheless, it is not necessary to compute this intersection; instead, the estimation procedure can test if the sensed configuration belongs to all the domains DC of the basic contacts involved in the multi-contact state.

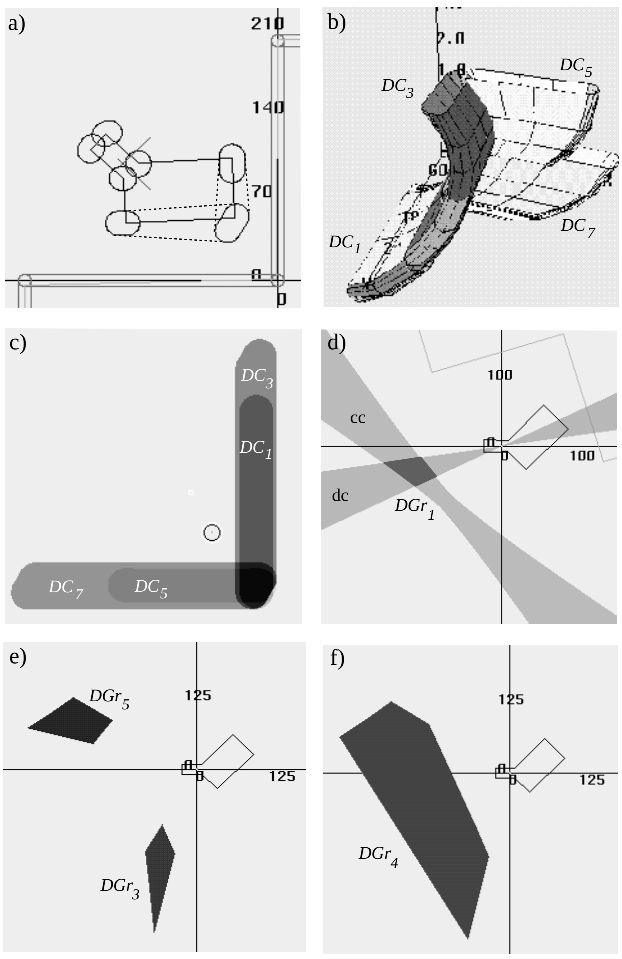

Fig. 3. *Physical space and configuration and force observation domains for the block-in-the-corner assembly task.*

Force Observation Domains

The algorithm to determine the generalized force realization domains DGr and the classification procedure make use of the dual representation of forces [4]. This representation maps a generalized force $[f_x, f_y, \tau]$ into a point $[\frac{f_y}{\tau}, -\frac{f_x}{\tau}]$ representing the force direction and a sign representing the force sense. In this way, a generalized force is represented by a sign and a point in a dual plane. Since the module of the reaction force is not relevant for the state estimation, it does not matter if it is not considered in the dual representation.

The domain of possible reaction forces DGr for a state with one basic contact is computed as the intersection, in the dual plane, of the two sets of points representing the forces that satisfy the following two conditions in the physical space:

- *Contact-point condition*: the line of force must intersect the region where the contact vertex can lie due to uncertainty.
- *Direction condition*: the reaction force direction must lie inside the friction cone enlarged with the uncertainties that affect the direction normal to the contact edge in the force reference frame attached to the gripper.

Figure 3d shows the dual representation of the domain DGr (dark shaded) for a state with only one basic contact (state 1 in figure 2) for a block orientation; it is obtained as the intersection of the two light shaded regions of dual points representing the forces that satisfy the contact-point condition (cc) and the direction condition (dc).

For states with more than one basic contact, the domain DGr is computed as a linear combination of the corresponding reaction forces at each basic contact involved. This can be performed by using geometric operations in the dual plane. A detailed description of the generation of the domains DGr can be found in [17]. As an example, figure 3e shows the domains DGr for two states for a block orientation, both with only one basic contact (states 3 and 5 in figure 2), and figure 3f shows the domain DGr for a state involving both basic contacts simultaneously (state 4 in figure 2).

The domains DG did not need to be explicitly computed off-line, the uncertainty of the force/torque sensor being directly included in the on-line estimation procedure.

2.3. On-line estimation

Configuration Observation Domains

As the boundaries of the domains DC are parametrized in the orientation of the moving object, testing if the current sensed configuration (x_o, y_o, ϕ_o) belongs to a given domain DC is equivalent to verifying if the point (x_o, y_o) lies inside the section $\phi = \phi_o$ of DC (for example, the current configuration of the block in figure 3c does not belong to any domain DC). This test only requires simple geometric operations in the plane, thus allowing to satisfy the time requirements for on-line computation.

Force Observation Domains

The domains DG are not explicitly built; instead, the classification procedure takes into account the force measurement uncertainty domain U_g, by testing in the dual plane which domains DGr overlap with U_g. Assuming that the force/torque sensor supplies independently the components of the generalized reaction force, U_g will be a parallelepiped, and the algorithm to verify if the direction of the measured force lies inside a domain DG will sequentially test if the dual representations of U_g and the corresponding domain DGr satisfy one of the following ordered conditions: a) a vertex of U_g is inside DGr, b) an edge of U_g crosses DGr, c) a face of U_g contains DGr. The classification procedure (detailed in [3]) requires, as in the configuration case, only simple geometric operations in the plane, thus being compatible with on-line time requirements.

3. LEARNING APPROACH

3.1. Methodology

This section presents an inductive learning approach to contact estimation. The main requirement of such an approach is the availability of examples to learn from. Examples of correct estimations can be obtained either by task executions or from a simulator (section 3.2). Then, the estimator can be trained to predict the contact state. In this work a backpropagation neural net is used as an estimator. One of the relevant properties of this type of networks is the universal approximation [11], which states that with enough but finite number of hidden neurons, there exists always a set of weights such that the network can approximate any nonlinear function to the desired accuracy. Since any estimation problem can be considered as a multi-input multi-output function approximation problem, this property virtually provides a theoretical foundation for using a backpropagation net to solve the problem of contact estimation. A backpropagation net features a layered structure and has weighted feedforward connections only between neurons in the adjacent layers. It is composed of a layer of input nodes, one or several hidden layers of neurons and an output layer of neurons. Each neuron in the network takes, as the input, the sum of the weighted outputs from other neurons connected to it, and then passes the value through a nonlinear function. Typical examples for such functions are a sigmoid and a tangent hyperbolic function.

3.2. Off-line computation

Sample generation

The state-samples are composed of three configuration data (x, y, ϕ), three force data (f_x, f_y, τ) and the corresponding task state label number.

In order to generate the configuration data, random values of the deviations of all the parameters subject to uncertainty are first chosen with uniform probability density function, and the corresponding actual C-surfaces for all the basic contacts are determined. Then, depending on the number of basic contacts considered, a configuration over a C-face, a C-edge, or directly a C-vertex, is equi-probably chosen.

Once a contact configuration has been determined, a random reaction force is also equi-probably selected within all the possible reaction forces compatible with the contact configuration. First, the direction and the sense of the generalized reaction force are determined; then, the module is chosen within a predefined range.

Since a priori the probability distribution of the state occurrence is not known, the same number of samples are generated for each state.

It is important in this phase to generate a large and complete set of examples, as typically the extrapolation capability of neural networks is very poor. For the selected problem, a training set of 1000 examples was appropriate. Using fewer than 300 examples produced poor results.

Network selection and training procedure

The neural network has 6 inputs (configuration and force components) and 9 outputs (one for each state). The classical backpropagation algorithm, modified with momentum and a variable learning rate [20], was used. The networks were initialized with the Nguyen-Widrow initialization method [15]. Various network topologies were trained in order to find a network with the best generalization capability. An appropriate number of hidden neurons is very important, since too small networks cannot make complex classifications, and too large ones will overfit the training data, which results in poor performance on independent test data. MATLAB's neural toolbox [7] has been used for this part of the experiment. The best results were obtained by using 15 hidden neurons, hyperbolic tangent activation functions in the hidden layer, and linear output neurons. In this case, the misclassification rate on an independent test set accounts for 8%. Figure 4 shows the misclassification rate on training and test sets during the learning process.

3.3. On-line estimation

Following the learning approach, on-line state estimation consists of a single evaluation of the network and the competitive layer, as shown in figure 5. The competitive layer selects the state corresponding to the output closest to one. Following the learning approach misclassifications can occur, the confusions between non-contiguous states being

	1	2	3	4	5	6	7	8	9
1	.	c	n	n	n	n	n	c	c
2	c	.	c	n	n	n	n	n	c
3	n	c	.	c	n	n	n	n	c
4	n	n	c	.	c	n	n	n	c
5	n	n	n	c	.	c	n	n	c
6	n	n	n	n	c	.	c	n	c
7	n	n	n	n	n	c	.	c	c
8	c	n	n	n	n	n	c	.	c
9	c	c	c	c	c	c	c	c	.

%	1	2	3	4	5	6	7	8	9
1	90.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
2	6.7	100.0	0.0	0.0	0.0	0.0	0.0	6.7	0.0
3	0.0	0.0	100.0	3.3	0.0	0.0	0.0	0.0	0.0
4	0.0	0.0	0.0	90.0	0.0	0.0	0.0	0.0	13.7
5	0.0	0.0	0.0	0.0	86.7	0.0	0.0	0.0	0.0
6	0.0	0.0	0.0	6.7	13.3	100.0	0.0	0.0	6.9
7	0.0	0.0	0.0	0.0	0.0	0.0	96.5	3.3	0.0
8	3.3	0.0	0.0	0.0	0.0	0.0	3.5	90.0	3.5
9	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	75.9

Fig. 6. *Left: state contiguity matrix ; 'c' for contiguous states and 'n' for non-contiguous ones. Right: the confusion matrix of a trained network for an independent test set, with 15 hyperbolic tangent hidden neurons and 9 linear output neurons. Columns represent the true states and rows the classifier decision. Classifications are reported in percentages.*

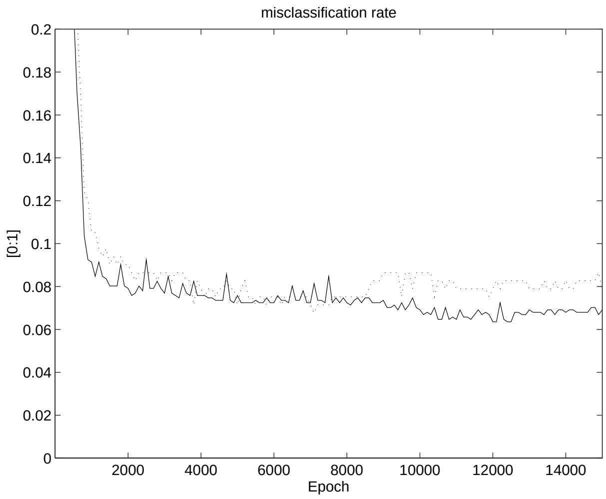

Fig. 4. *Evolution of the misclassification rate on the training set (full line) and on the test set (dotted line) during the training of a backpropagation network.*

more serious than between contiguous states[1]. The reason for this is that the transition operators of contiguous states are likely to make the task evolve in a similar way, and therefore the confusion has not so bad consequences. Hence the learning algorithm should mainly try not to confuse non-contiguous states. Confusion matrices, as shown in figure 6, can be useful in order to visualize the misclassifications between states.

4. APPROACHES COMPARISON

Figure 7 shows a block diagram summarizing the analytical and the learning approaches and the parallelism between them. The following is a summary of the advantages and disadvantages of each approach.

[1] Two states are contiguous when it is possible to pass from one to the other without the ocurrence of any other state [19].

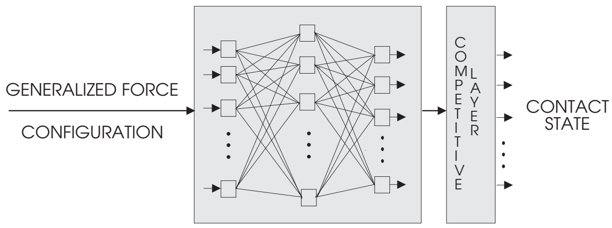

Fig. 5. *A neural net is used to estimate the contact state. The competitive layer selects the neuron with the highest output. Each output neuron represents a contact state.*

4.1. Analytical Approach

Advantages:

- Covers all possible situations provided that everything is well modeled.
- The estimated state is always correct provided that the classification is unambiguous.
- The computational cost of changing the assembly task is low because the equations used by the algorithms are parametrized in the objects' geometry.

Disadvantages:

- When there is more than one contact situation compatible with the sensed data, the method cannot decide which is the correct one. It would be necessary to apply some heuristics to perform the decision.
- It is difficult to apply to 6 d.o.f.
- On-line time constraints may become a problem.

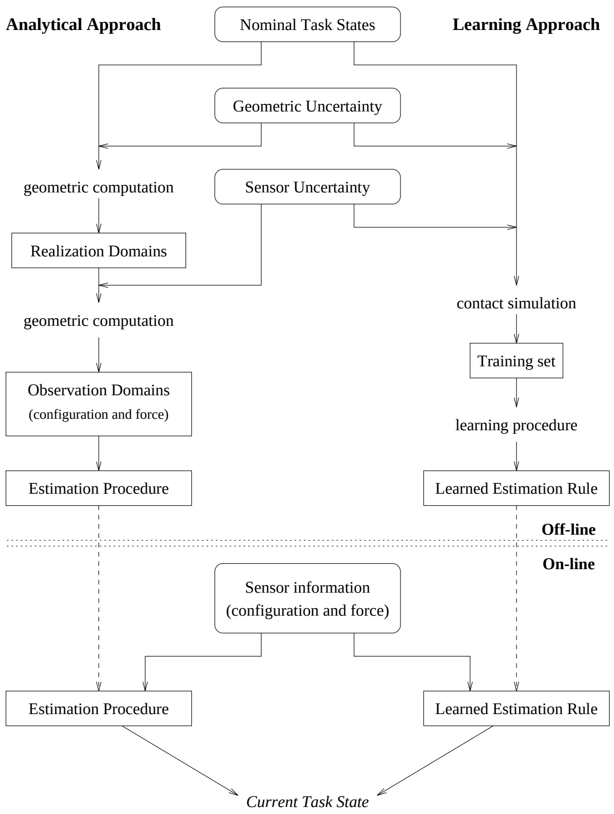

Fig. 7. *Comparison between analytical and learning approaches.*

4.2. Learning Approach

Advantages:

- When the examples are taken from real experiments (non-uniform distributions) they capture all the real uncertainties (geometric uncertainty and sensor uncertainty) as well as their probabilistic distribution within the observation domains, giving rise to an efficient partition of the configuration and force spaces.
- Simple and quick on-line estimation.
- More easily extendable to 6 d.o.f.

Disadvantages:

- Necessity of a training set: if the training examples are taken from real experiments, it is difficult to reproduce a representative set of all the possible situations; if the training examples are taken from simulation, the proper selection of the samples distribution is not clear.
- Misclassifications are possible; then, there is no guarantee of the correctness of the estimated state.
- The estimation rule must be re-learned for each assembly task.

5. CONCLUSIONS

The estimation of the current contact situation during an assembly task is necessary for the determination of the proper robot movement in order to follow an assembly plan. Two approaches to contact estimation have been presented: an analytical approach and an inductive learning approach. The analytical approach studies all the uncertainties affecting an assembly task and computes the sets of configurations and forces that are compatible with each contact situation. The current contact situation is then estimated by classifying the measured configuration and force into one of these sets. The inductive learning uses examples of contact situations generated by a simulator to train a backpropagation neural net. Once trained, the contact estimation is simply an evaluation of the network.

The analytical approach can determine with certainty the task states compatible with the measured data though, when more than one state is compatible, it needs some heuristics to select one of them. The learning approach is computationally simpler and has a shorter on-line estimation time, but the estimation rule must be learned for each assembly task, whereas changing the task is easier for the estimation procedure of the analytical approach. Both approaches rely on models which are used, in the analytical case, for geometric reasoning and, in the learning case, for the generation of simulated examples to have a proper training set.

Acknowledgements

The authors would like to thank A. Giordana and M. Kaiser for their support.

6. REFERENCES

[1] C. Baroglio, A. Giordana, M. Kaiser, M. Nuttin, R. Piola, Learning Controllers for Industrial Robots, *Technical Report UTO, K.U.Leuven, UKA*, 1994.

[2] L. Basañez and R. Suárez, Uncertainty Modelling in Configuration Space for Robotic Motion Planning, *Proc. of the SYROCO'91*, Vienna, Austria, pp. 675-680, 1991.

[3] L. Basañez, R. Suárez and J. Rosell, Contact Situations from Observed Reaction Forces in Assembly with Uncertainty. *Technical Report IC-DT 9412*, 1994.

[4] R. C. Brost and M. Mason, Graphical Analysis of Planar Rigid-Body Dynamics with Multiple

Frictional Contacts, *Robotics Research: The Fifth Int. Symp.*, 1989.

[5] S.J. Buckley, Planning and Teaching Compliant Motion Strategies, MIT Artificial Intelligence Lab., report AI-TR-936 (Ph.D Thesis), 1987.

[6] G. Dakin, Fine-motion planning for robotic assembly in local contact space. Ph.D. Thesis, Univerity of Massachusetts, Amherst, MA, 1994.

[7] H. Demuth and M. Beale. *Neural Network Toolbox.* The Math Works Inc., Cochituate Place, 24 Prime Park Way, Natick, Massachusetts 01760, 1994.

[8] S.N. Gottschlich S.N. and A.C. Kak, Motion Planning for Assembly Matting Operations. *Proceedings of the 1991 IEEE International Conference on Robotics and Automation*, pp.1959-1963.

[9] V. Gullapalli, A. G. Barto and R. A. Grupen, Learning Admittance Mappings for Force-guided Assembly.. *Proceedings of the 1994 IEEE International Conference on Robotics and Automation*, pp.2633-2638.

[10] S. Hirai and K. Iwata, Derivation of Damping Control Parameters Based on Geometric Model. *Proceedings of the 1993 IEEE International Conference on Robotics and Automation*, pp. 87-92.

[11] K. Hornik, M. Stinchcombe and H. White, Multilayer feedforward networks are universal approximators, *Neural Net.*, Vol.2, pp.359-366, 1989.

[12] C. Laugier, Planning Fine Motion Strategies by Reasoning in Contact Space. *Proceedings of the 1989 IEEE International Conference on Robotics and Automation.*

[13] T. Lozano Perez, M. Mason and R. Taylor, Automatic Synthesis of Fine-Motion Strategies for Robots, *The Int. Journal of Robotics Research*, 3 (1), pp. 3-24, 1984.

[14] M. Nuttin, H. Van Brussel, C. Baroglio and R. Piola, Fuzzy Controller Synthesis in robotic assembly: Procedure and experiments. *FUZZ-IEEE94: Third International Conference on Fuzzy Systems, World Congress on Computational Intelligence, 1994.*

[15] D. Nguyen and B. Widrow, Improving the learning speed of 2-layer neural networks by choosing initial values of the adaptive weights, *International Joint Conference of Neural Networks*, volume 3, pages 21–26, July 1990.

[16] J. M. Schimmels and M. A. Peshkin, Synthesis and Validation of Non-Diagonal Accommodation Matrices for Error-Corrective Assembly. *Proceedings of the IEEE 1990 International Conference on Robotics and Automation*, pp.714-719.

[17] R. Suárez, L. Basañez and J. Rosell, Assembly Contact Force Domains in the Presence of Uncertainty, *Proc. of the SYROCO'94*, Capri, Italy, 1994.

[18] R. Suárez and L. Basañez, Assembly With Robots in Presence of Uncertainty, *Proc. of the 22nd ISIR*, Detroit, USA, pp. 19/1–19/15, 1991.

[19] R. Suárez and L. Basañez, Fine Motion Planning in Presence of Uncertainty, *II European Workshop on Learning Robots*, Torino, Italy, pp.143-164 ,1993.

[20] T.P. Vogl, J.K. Mangis, A.K. Rigler, W.T. Zink and D.L. Alkon, Accelerating the convergence of the backpropagation method, *Biological Cybernetics* Vol. 59, pp. 257-263, 1988.