Proceedings of the 2000 IEEE
International Conference on Robotics & Automation
San Francisco, CA • April 2000

Evolution-Based Virtual Training in Extracting Fuzzy Knowledge for Deburring Tasks

S.-F. Su [1], T.-J Horng [2], and K.-Y. Young [2]

[1] Department of Electrical Engineering
National Taiwan University of Science and Technology
Taiwan R. O. C.

[2] Department of Electrical and Control Engineering,
National Chiao Tung University
Taiwan, R. O. C.

Abstract

In this research, the problems of how to teach a robot to execute skilled operations are studied. Human workers usually accumulate his experience after executing the same task repetitively. In the process of training, the worker must find ways of adjusting his/her execution. In our system, the parameters for the impedance control scheme are used as the targets for adjustment. After mass amount of training, the worker is supposed to be able to execute deburring tasks successfully. This is because the worker might have gotten some knowledge about tuning the parameters required in the impedance control scheme. Thus, the rules for adjusting the parameters in impedance control are the operational skills to be identified. In this research, a training scheme, called the evolution-based virtual training scheme, is proposed in extracting knowledge for robotic deburring tasks. In this approach, a evolution strategy is employed to searching for the best set of fuzzy rules. This learning scheme has been successfully applied in adjusting the parameters of impedance controllers required in deburring operations. In general, the results of deburring are much satisfactory when compared with those in the previous research. When executing a deburring task, the robot simulator can find its optimal adjusting rules for parameters after several generations of evolution.

1. Introduction

Recently, industry has successfully used robots in engaging in executing various tasks whose working environment is harmful to human beings or whose operations are repetitive and/or require high accuracy. Usually, those tasks can be programmed into the operations of robots because those tasks do not interact with the environment frequently and then human skill may not be necessary for the operations of the tasks. On the other hand, there exist tasks, such as deburring, grinding, milling, assembly, etc., which may need a great deal of interactions with the environment and thus, require lots of decision-making processes while facing those interactions. Hence, the successful execution of those tasks largely relies on human skill in achieving satisfactory results. Such kinds of tasks are very difficult to be satisfactorily programmed into the operation of robots. In fact, even there exists some work that has tried to manage to embed those tasks into the operation of robots, those operations may face lots of problems when uncertainty occurs in the environment and the actual operations may not be satisfactory.

Several researchers have tried to discover the relationships between human experts' intentions and operational strategies for tasks so that the skills could be modeled accordingly and then are possibly transferred to the operations of robots. Asada *et al.* have tried to use neural networks [1,2], adaptive control [3], or fuzzy rules [4] to model and to transfer human skills to the operations of robots. In [5], in order to acquire human skills, the authors have relaxed the joints of a robot manipulator, and let a human expert worker take the end-effector of a manipulator to accomplish a compliant task, for instance, deburring. The data of the deburring process, such as positions (angles) and forces (torques) of all joints are recorded. The approach is then to extract useful rules or strategies for representing the skills from the collected data. However, some problems may arise in the above approach. First, the obtained rules or strategies based on this set of data may not be able to represent the skills in sufficiency. When an expert worker need to take the end-effector to proceed the execution, since in this unusual way the worker cannot execute the task as he/she usually did, the results may not be satisfactory. Besides, the recorded data may not be sufficient to adequately express the operational skills. Secondly, the rules or strategies extracted from the data are rather primitive and are sensitive to the operational conditions. For example, in deburring tasks, if different material properties or surface roughness of workpieces are considered for deburring tasks, the obtained strategies may not work well. Therefore, it may be required to further generalize that obtained knowledge to cope with the variation of tasks. Thus, in this research we attempted to propose another way of defining operational skills and training schemes for learning the deburring tasks.

In the work of [11], human skills are stored in the desired position commands to the controller of a robot manipulator. In that approach, other parameters, for deburring tasks, the parameters required in the impedance control scheme are chosen as constants. However, the results in our simulation have shown that the performance can be improved by changing the parameters in the impedance control mechanism. In order to obtain better results for deburring operations, a way of determining when and how to change those parameters must be defined. In this research, a fuzzy rule decision-making system is proposed to define strategies for changing those parameters. In other words, we attempted to use the parameters to represent the operational skills for deburring. However, it is difficult to obtain such rules from expert knowledge or from training data. Thus, we proposed another concept of learning called virtual training for our study.

2. The Concept of Virtual Training

The idea of virtual training is to check the results of the operations shown in a virtual environment, which is defined as a simulation system that can truly model the considered plant. When the results are not good enough, the operational schemes are changed according to some criteria, and another simulation is then conducted again. Hopefully, after a period of training, the operational scheme yielding acceptable performance can be obtained.

From the training point of view, ideally, a worker is trained to know how to satisfactorily execute deburring tasks, and such knowledge is accumulated from repetitive operations. With a virtual environment, human operators can virtually execute the tasks to obtain training data. The first advantages of using virtual training is that since the operations of tasks are emulated in the simulation, there is no cost for repetitive operations, and then mass amount of training become possible. Another fold of advantages is that it is possible to capture various kinds of information or states that may be difficult or impossible to get in the real world operations.

In the process of training, the worker must find ways of adjusting his/her execution. In our system, the parameters for the impedance control scheme are used as the targets for adjustment. After mass amount of training, the worker is supposed to be able to execute deburring tasks successfully. This is because the worker might have gotten some knowledge about tuning the parameters required in the impedance control scheme. Hence, the learning task is to analyze the data recorded in the operation process and to extract the operational skills from those data. In this research, the analysis is to identify the relationship between the parameters and the performance of the operation and to estimate what values for those parameters should be set for certain situations. Naturally, such a kind of knowledge is represented in a rule base. In our work, the knowledge obtained from the collected data is also converted into rules. However, the representation of knowledge by using rules is discrete in nature, but the data obtained are all continuous variables. Therefore, in order to soften the boundary between rules, the fuzzy concept is included into the rule reasoning system for the operational skills.

In a training paradigm, various patterns must be used to explore all possibilities to search for the pattern that can yield the best performance. However, if the training patterns are tried in a random manner, the search for finding the best candidate may take lots of time, or even worse may not find a feasible solution. For our case, the number of possible rule sets is extremely large, and then the exhaustive search for all rule sets is impossible. Therefore, it is nature to think about the use of evolution strategies to provide an effective way of finding the best candidate. It will be introduced later. In this paper, an evolution based virtual training scheme used to find rules for determining proper parameters in impedance control is proposed.

3. Impedance Control

In this research, deburring tasks are the skilled operations for training. Deburring tasks are usually carried out by impedance control mechanisms [6,17-19]. The models used by Asada *et al.* in [6] and by Kazerooni *et al.* in [17-19] are all impedance control schemes. In [6], the authors considered that the function of the operation should be simple enough so that it does not require heavy computation. Nevertheless, the function must also model the motion executed by human experts with a reasonable accuracy. The authors in [17-19] have also chosen impedance control with a view of the specifications of performance and robustness. Thus, in this work, impedance control is also employed as the compliance controller for deburring operations.

In an impedance control scheme, with simple physics, the following control is used:

$$M_d\ddot{X} + B_d\dot{X} + K_d(X - X_d) = F_{ext}, \qquad (1)$$

where M_d, B_d and K_d are the inertial, the damping and the stiffness matrix, respectively, specified by the designer. X, $\dot{X}$, and $\ddot{X}$ are the end-effector's position, velocity and acceleration vectors, respectively. X_d is the goal position vector in the Cartesian space, and F_{ext} is the external force caused by the environment. The impedance in Eq. (1) is a decoupled form in the Cartesian space. Since the control of robot manipulator is always considered in the joint space, the above equations must be transformed into the joint space to derive the control torque. From the robot dynamics [9] and straightforward manipulation, the control law is obtained. The detailed discussions can be seen in [14].

In the use of impedance control, one major issue must be resolved. It is about how to choose the suitable target impedance or those parameters in Eq. (1). In the work of [11], those parameters are chosen as constants in his work of transferring human skills. However, the results in the later simulation have shown that the performance can be improved by changing the parameters in the impedance control mechanism. Thus, in order to obtain a better result for deburring operations, a way of determining when and how to change those parameters must be defined.

4. Fuzzy Rules for Deburring Tasks

Asada *et al.* [4] had ever made use of fuzzy rules as the skills in robotic deburring operations. In the approach, the moving trajectory of a human expert is measured and then categorized based on the expert linguistic information. An example of the expert linguistic information is that if a burr is large, the cutting force should be increased in general. The linguistic control rules are constituted of both discriminate functions and a group of associate linear mappings. Those functions are identified from the obtained data.

In this research, the above concept of fuzzy rules is adopted to capture the skills required in deburring operations. As we have stated, in this work, the rules for adjusting the parameters in impedance control are the operational skills to be identified. Thus, the training is to identify the rules that imitate the behavior of human experts to increase or to decrease the external force in the impedance controller when the error between the current point X and the desired position X_d is large. If the end-effector has stayed at the same position for a long time, it means that the X_d command must be changed to move the end-effector ahead again.

According to the above description, the rules for adjusting the parameters of impedance controllers are of the form "*IF* (A_{li} < *error* < A_{ui}) *THEN* (ΔM, ΔB, ΔK, ΔX_d) = P_i". In the rule, A_{li} and A_{ui} are the lower and the upper boundaries, respectively, for the i-th rule. ΔM, ΔB, ΔK, and ΔX_d are the variables for the changes, respectively, of M_d, B_d, K_d, and X_d in Eq. (1) to be set. P_i is a set of values for ΔM, ΔB, ΔK, and ΔX_d of the i-th rule. Thus, P_i is the target to be learned. Note that since only line deburring tasks are discussed, the dimensions for the above variables are all 2. It is worthy mentioning that since the training is to determine what is P_i in the rules, in the rule structure, P_i is a set of crisp values instead of fuzzy sets. Thus, the simplified TSK [8] reasoning mechanism is used in our work.

5. Evolution Strategy

Evolution strategy is a way of finding parameters that optimize an objective function [7]. In fact, evolution strategies are often referred as genetic algorithms, which are defined for the problems of finding the best combination of gene [13]. Evolution strategies have often been utilized in solving a wide range of optimization problems. Various applications can be found in robotics, such as solving inverse kinematics of redundant robots [21], motion planning and obstacle avoidance [22], or navigation of mobile robot [12]. They have also be employed in fuzzy systems or neural networks to search for the optimal answer [10,15,20].

In this work, the forms of the initial chromosomes include eight attributes, ΔM_1, ΔM_2, ΔB_1, ΔB_2, ΔK_1, ΔK_2, ΔX_{1d}, and ΔX_{2d}. Those values are to be searched through training. An example of an initial chromosome is shown in Figure 1. In reproduction operations, the chosen probabilities of chromosomes depend on their fitness value. A higher evaluation score results in a higher chosen probability. Floating crossover operation is employed. For instance, consider two chromosomes, $\zeta = <\zeta_1, \zeta_2, \cdots, \zeta_n>$ and $\eta = <\eta_1, \eta_2, \cdots, \eta_n>$. A crossover site i is selected at random. In our approach, new offspring are obtained as $<\zeta_1 + (\eta_1 - \zeta_1)/a_1, \cdots, \zeta_n + (\eta_n - \zeta_n)/a_n>$ and $<\eta_1 + (\zeta_1 - \eta_1)/b_1, \cdots, \eta_n + (\zeta_n - \eta_n)/b_n>$, where a_i and b_i are real numbers greater than 1, for $i=1 \ldots n$. In the simulation, a_i and b_i are set between 2 and 5.

Mutation value must be limited to a given range; otherwise, the operation may become unstable. Let ζ_i in $\zeta = <\zeta_1, \zeta_2, \cdots, \zeta_i, \zeta_{i+1}, \cdots, \zeta_n>$ be the chosen attribute to be mutated. Then it is directly as $\zeta_i + \delta$ where δ is a real number with given the reasonable extension. In our simulation, each attribute has its individual mutation probability. Then, there may be more than one attribute that will mutate in a chromosome. The mutation probability of each attribute is 0.03 and δ is limited within the region [-3,3]. The initial population of those impedance parameters is shown in Tables 1.

6. System Implementation and Simulation Results

Figure 2 shows the block diagram of the proposed learning system. First, the rules (initial population of the evolution strategy) enter into the system and are soften with the fuzzy module. Then, the impedance controller use those soften rules to adjust its parameters during the task execution. Note that a free space trajectory must be planed in advance, because the robot is asked to move in the free space automatically. The same initial situations are needed to make meaningful comparisons. Thus, the trajectory in the free space and the contact point of each simulation cycle are fixed. When a deburring simulation task has been accomplished, the fitness of the set of rules is evaluated according to the results. The evolution strategy is then used to produce new offspring (new sets of rules). Hopefully, with the above evolution-based training mechanism, a satisfactory set of rules can be acquired.

The dynamic model proposed in [11] is adopted in this simulation. The dynamics of the working environment is described as:

$$-F_{ext} = B_e \dot{X} + K_e (X - X_e)$$
$$F_{fri} = \mathrm{sgn}(v_{\tan}) C_{fri} F_{nor}$$
$$\dot{X}_{e1} = -k_b \frac{F_{ext1}}{|\dot{x}_2| + \sigma}$$

where F_{ext} is the external force caused by the environment, F_{fri} is the friction, F_{nor} is the force normal to the contact surface, C_{fri} is the coefficient of friction, $v_{\tan}$ is the velocity of the tangent direction, and

k_b and σ are positive constants. The external force is caused by the deformation of the workpiece, and may debur the extra material beyond the desired line. Due to the contact force, the position of the surface of the workpiece will approach to the desired surface. In the differential equation describing the deburring process, the constant k_b represents the efficiency of deburring. When k_b is large, the efficiency of deburring is high. A small positive constant σ is used as a bias term to avoid dividing by zero in the equation. In addition to the normal contact force, the velocity of the horizontal direction in the deburring process is also related to the vertical velocity of the end-effector of the manipulator.

In this simulation, a set of feasible rules for adjusting the parameters required in an impedance controller must be acquired at first to define the initial population for evolution-based training. The requirements for those rules are to make sure that the whole system be stable during the deburring operation. In order to soften the boundary between rules with the fuzzy concept, the mechanism of approximate reasoning is employed to interpreting those rules. Hopefully, after training, some usable strategies as in our case, rules for adjusting parameters, M, B, K, and X_d, to accomplish a deburring task with satisfaction can be extracted.

The requirements of deburring tasks in our training are described as follows. First, when the end-effector of the robot manipulator is in contact with the working environment, big impact should be avoided. Secondly, the line after deburring is asked to be as close to the desired line as possible, and then the position error is one index of the performance measure. Thirdly, the stability during the execution of the task needs to be assured. Finally, the execution time should not exceed a prescribed value. Nevertheless, the weight of the position performance in the evaluation score is larger than that of the time consumption and that of the force exertion. If the robot simulator can complete the task fast and the results can match the task requirements, it is perfect.

According to the above task requirements, the evaluation function used in our work to evaluate the performance of a deburring operation is defined as:

$$Score = \frac{W_1 \times score_1}{2 \times count} + \frac{W_2 \times score_2}{2 \times count} + (100 - W_1 - W_2) \times \exp(-p_2 \times count)$$

$$score_1 = \sum_{i=0}^{n} [\operatorname{sgn}(0.4 - X_{01i}) \times \exp(-p_3 \times (0.4 - X_{01i}))]$$

$$score_2 = \sum_{i=0}^{n} [\exp(-p_1 \times (F_{ext1} - F_{lmt})^2) \times \exp(-p_1 \times (F_{ext2} - F_{lmt})^2)]$$

where F_{lmt} is the desired force. W_1 and W_2 are weights of the position score and of the force score, respectively. p_1, p_2, and p_3 are the multipliers of each score. 0.4 is the desired position and X_{01i} is the position after deburring. $count = n$ is the number of points needed to accomplish a deburring task and is the measure of the execution time. In this simulation, the counting number can not exceed a desired number. If it exceeds this desired number, the evaluation score is set as zero. In other words, if the moving speed of the deburring operation is too slow, the deburring operation will be regarded as a fail operation. In the same way, if the external force exceed the tolerable force, the robot simulator alerts the user and also the evaluation score is set as zero. Those parameters and the weights used in our simulations are given in Table 2.

First, the simulation results using the initial rules are presented. The used environment parameters are B_{e1}=0.5, B_{e2} = 0.5, K_{e1} =100000, and K_{e2}=10000. The results are as shown in Figure 3. For comparison, the results in [11] are also cited and shown in Figure 4. It can be found that our results are much better than those in [11]. The results of applied the rules obtained from our evolution-based virtual training are shown in Figures 5-7. Those results are obtained with different burr sizes but with the same environmental stiffness. From the results, the performance is better than that by using the initial rule set. It can be concluded that this approach can indeed achieve the purpose of training. If the moving velocity of X_2 dimension keeps a low-valued constant, the line after deburring will be smooth. In addition, if the exerting force is sufficiently large, the line will be close to the desired line. If the moving speed is too large or not a constant, the line after deburring will be rugged and rough. Moreover, it may destroy the workpiece or the robot environment.

For the workpiece with the same stiffness but different burr sizes, the robot simulator can find its proper target impedance. The results are shown in Figures xx-xx. The obtained rules can also be successfully applied to the workpieces with the same environmental parameters but different bur sizes. The set of the rules obtained from the first case is used to debur the workpiece of the second case. Similarly, the set of the rules obtained from the second case is also used to debur the workpiece of the first case. The simulation results are shown in Figures 8 and 9.

7. Conclusions

In this research, the problems of how to teach a robot to execute skilled operations are studied. The idea of transferring the human skills to robots is somewhat different from those proposed in the literature. Even a human expert meet a new material that he never faced before, he may treat this material with his experience and intuition. Human workers usually accumulate his experience after executing the same task repetitively. In addition, human experts usually teach other workers how to execute a skilled task by showing the executing process, instead of expressing the skills by some sort of information or data. For instance, if the burr size is larger, then the exerting force should be larger. Such knowledge is difficult to be represented both sufficiently and accurately no matter in numerical data or in linguistic rules. As a consequence, the skills required in a skilled operation are difficult to be transferred to the operation for a robot. In this research, a new training scheme, called the evolution-based virtual training scheme, was proposed in extracting knowledge for robotic deburring tasks.

The results in our simulation have shown that the performance can be improved by changing the parameters in the impedance control mechanism. A fuzzy rule decision-making system was proposed in this research to define strategies for changing those parameters. In order to obtain a better result for deburring operations, a way of determining when and how to change those parameters must be defined. The number of possible rule sets is extremely large, and then the exhaustive search for all rule sets is impossible. In this paper, an evolution based virtual training scheme used to find rules for determining proper parameters in impedance control is proposed. This learning scheme has been successfully applied in adjusting the parameters of impedance controllers required in deburring operations. In general, the results of deburring are much satisfactory when compared with those in the previous research. When executing a deburring task, the robot simulator can find its optimal adjusting rules for parameters after several generations of evolution.

8. Reference

[1] H. Asada and S. Liu, "Transfer of Human skills to Neural Net Robot Controllers," *IEEE Int. Conf. Robotics and Automation*, pp. 2442-2448, 1991.

[2] H. Asada and S. Liu, "Teaching and Learning of Deburring Robots Using Neural Networks," *IEEE Int. Conf. Robotics and Automation*, pp. 339-345, 1993.

[3] H. Asada and S. Liu, "Adaptive Control of Deburring Robots Based on Human Skill," *IEEE Int. Conf. Decision and Control*, pp. 348-353, 1991.

[4] H. Asada and B. H. Yang, "Hybrid Linguistic/ Numeric Control of Deburring Robots Based on Human Skills," *IEEE Int. Conf. Robotics and Automation*, pp. 1467-1474, 1992.

[5] H. Asada, K. Kosuge, and T. Fukuda, "Acquisition of Human Skills for Robotic Systems," *IEEE Int. Symposium on Intelligent Control*, pp. 469-474, 1991.

[6] H. Asada and Y. Asari, "The Direct Teaching of Tool Manipulation Skills Via the Impedance Identification of Human Motions," *IEEE Int. Conf. Robotics and Automation*, pp. 1269-1274, 1988.

[7] T. Bäck, U. Hammel, and H.-P. Schwefel, "Evolutionary computation: Comments on the History and current state," *IEEE Trans. On Evolutionary Comp.*, vol. 1, no. 1, 1997.

[8] C. T. Lin and C.S.G. Lee, *Neural Fuzzy System: A Neuro - Fuzzy Synergism to Intelligent Systems*, Prentice-Hall, 1996.

[9] K. S. Fu, R.C. Gonzalez, and C.S.G. Lee, *Robotics: Control, Sensing, Vision, and Intelligence*, McGraw-Hill Book Company, 1989.

[10] T. Fukuda and K. Shimojima, "Fusion of Fuzzy, NN, GA to the Intelligent Robotics," *IEEE Int. Conf. Syst., Man, and Cybe.*, pp. 2892-2897, 1995.

[11] C.P. Guan, *An intelligent Man-Machine Interface for Human Skill Transfer in a Virtual Compliant Environment*, Master Thesis, Dep. of Cont. Eng., Chiao-Tung University, Taiwan, R.O.C., 1995.

[12] H. Hashimoto, F. Harashima, and K.Daehee, "Path Generation for Mobile Robot Navigation using Genetic Algorithm," *IEEE 21st Int. Conf. Industrial Electronics, Control, and Instrumentation*, pp. 167-172, 1995.

[13] J. Holland, *Adaptation in Natural and Artificial Systems*, Univ. of Michigan Press, 1975.

[14] T. J. Hrong, *An Intelligent Human-Machine Interface for Robot/VR Integration*, Master. Thesis, Dep. of Cont. Eng., Chiao-Tung University, Taiwan, R.O.C., 1996.

[15] H. Ishibuchi, K. Nozaki, N. Yamamoto, and H. Tanaka, "Selecting Fuzzy If-Then Rules for Classification Problems Using Genetic Algorithms," *IEEE Trans. on Fuzzy System*, Vol. 3, no. 3, pp. 260-270, 1995.

[16] C. L. Karr, and E. J. Gentry, "Fuzzy Control of pH Using Genetic Algorithms," *IEEE Trans. Fuzzy Systems*, Vol. 1, no. 1, pp. 46-53, 1993.

[17] H. Kazerooni, P.K. Houpt, and T. B. Sheridan, "Robust Compliant Motion for Manipulators, Part I, II," *IEEE J. Robotics and Automation*, Vol. 2, no. 2, pp. 83-105, 1986.

[18] H. Kazerooni, J.J. Bausch, and B.M. Kramer, "An Approach to Automated Deburring by Robot Manipulators," *J. Dyn. Syst., Meas. Cont.*, Vol. 108, pp. 354-359, 1986.

[19] H. Kazerooni and M.G. Her, "Automated Robotic Deburring of Parts Using Compliance Control," *J. Dyn. Syst., Meas., Cont.*, Vol. 113, pp. 60-66, 1991.

[20] D. Park, A. Kandel, and G. Langholz, "Genetic-Based New Fuzzy Reasoning Models with Application to Fuzzy Control," *IEEE Trans. Syst., Man, and Cybe*, Vol. 24(1), pp. 39-47, 1994.

[21] J. K. Parker, A. R. Khoogar, and D. E. Goldberg, "Inverse Kinematics of Redundant Robots using Genetic Algorithms," *IEEE Int. Conf. Robotics and Automation*, pp. 271-276, 1989.

[22] T. Shibata and T. Fukuda, "Intelligent Motion Planning by Genetic Algorithm with Fuzzy Critic," *IEEE Int. Symposium on Intelligent Control*, pp. 565-570, 1993.

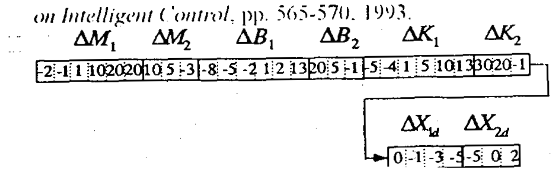

Figure 1. One example of the used initial chromosomes.

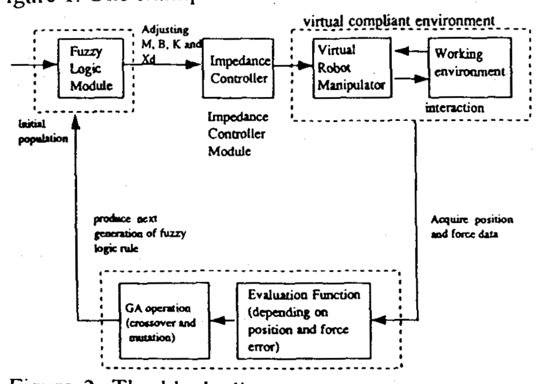

Figure 2. The block diagram of the proposed training system.

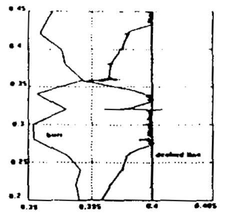

Figure 3. The deburring result for the initial rule.

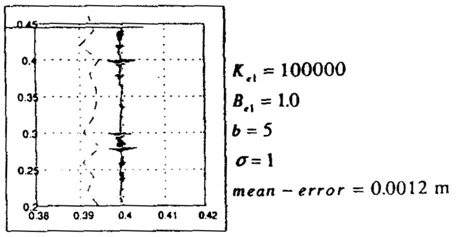

Figure 4. The deburring result presented in [11].

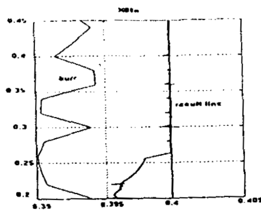

Figure 5. The deburring result of the first case.

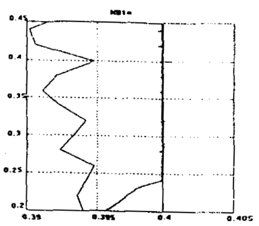

Figure 6. The deburring result of the second case.

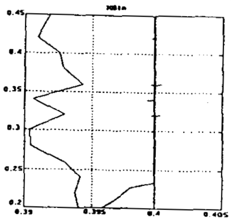

Figure 7. The deburring result of the third case.

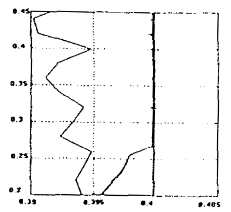

Figure 8. The deburring result of the second case by using the rules obtained for the first case.

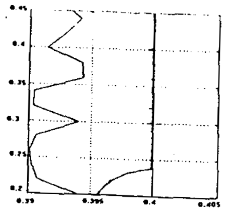

Figure 9. The deburring result of the first case by using the rules obtained for the second case..

Table 1. The initial population used in the study. (a) M_1, (b) M_2, (c) B_1, (d) B_2, (e) K_1, (f) K_2, (g) X_{1d}, (h) X_{2d}.

(a) M_1

Region	Ten initial chromosomes ($\times 5 \times 10^{-4}$)									
$Error < -5.0 \times 10^{-4}$	-2	-2	-3	-3	-3	-2	-2	-2	-5	-5
$-5.0 \times 10^{-4} \le Error < 0.0$	-1	-1	-1	-2	-2	-2	-2	-2	-2	-2
$0.0 \le Error < 10^{-3}$	1	1	1	1	1	1	1	1	1	1
$10^{-3} \le Error < 2.0 \times 10^{-3}$	10	8	7	9	9	8	8	6	6	6
$2.0 \times 10^{-3} \le Error < 3.0^{-3}$	20	12	15	11	11	15	15	14	14	12
$3.0 \times 10^{-3} \le Error$	20	13	20	16	16	18	18	20	20	15

(b) M_2

Region	Ten initial chromosomes ($\times 10^{-3}$)									
$X_2speed \ge 3.0 \times 10^{-2}$	10	8	7	7	6	6	5	10	7	7
$10^{-2} \le X_2speed < 3.0 \times 10^{-2}$	5	3	3	5	5	4	3	3	5	5
$X_2speed \le 10^{-2}$	-3	-1	-2	-3	-2	-2	-2	-2	-3	-3

(c) B_1

Region	Ten initial chromosomes ($\times 3 \times 10^{-2}$)									
$Error < -10^{-3}$	-8	-6	-5	-5	-5	-5	-10	-10	-10	-8
$-10^{-3} \le Error < -5.0 \times 10^{-4}$	-5	-4	-3	-4	-4	-4	-4	-4	-4	-5
$-5.0 \times 10^{-4} \le Error < 0.0$	-2	-1	-1	-2	-2	-2	-1	-1	-1	-2
$0.0 \le Error < 8.0 \times 10^{-4}$	1	1	1	1	1	1	1	1	1	1
$8.0 \times 10^{-4} \le Error < 1.5^{-3}$	2	2	2	3	3	2	2	2	2	2
$1.5 \times 10^{-3} \le Error$	13	10	10	9	9	8	5	5	6	8

(d) B_2

Region	Ten initial chromosomes ($\times 10^{-1}$)									
$X_2speed \ge 2.0 \times 10^{-2}$	20	15	12	12	10	10	20	18	11	20
$0.0 \le X_2speed < 2.0 \times 10^{-2}$	5	3	3	3	4	4	5	5	7	5
$X_2speed < 0.0$	-1	-1	-2	-2	-2	-2	-1	-2	-3	-1

(e) K_1

Region	Ten initial chromosomes									
$Error < -10^{-3}$	-5	-5	-5	-5	-4	-4	-5	-5	-5	-5
$-10^{-3} \le Error < 0.0$	-4	-4	-4	-4	-3	-3	-4	-4	-4	-4
$0.0 \le Error < 10^{-4}$	1	1	1	1	1	1	1	1	1	1
$10^{-4} \le Error < 3.0 \times 10^{-3}$	5	3	3	3	3	3	5	3	3	2
$3.0 \times 10^{-3} \le Error < 5.0^{-3}$	10	8	8	8	7	9	10	7	7	6
$5.0 \times 10^{-3} \le Error$	13	10	10	10	9	9	13	9	9	7

(f) K_2

Region	Ten initial chromosomes ($\times 10^{-4}$)									
$X_2speed \ge 2.5 \times 10^{-2}$	30	25	20	15	10	12	22	20	20	30
$1.5 \times 10^{-2} \le X_2speed < 2.5 \times 10^{-2}$	20	15	12	10	8	6	16	18	18	15
$X_2speed < 1.5 \times 10^{-2}$	-1	-2	-3	-4	-1	-1	-1	-1	-3	-3

(g) X_{1d}

Region	Ten initial chromosomes ($\times 10^{-3}$)									
$Error \ge 0.0$	0	0	0	0	0	0	0	0	0	0
$-10^{-4} \le Error < 0.0$	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1
$-5.0 \times 10^{-4} \le Error < -10^{-4}$	-3	-3	-2	-2	-2	-2	-1	-1	-1	-1
$Error < -5.0 \times 10^{-3}$	-5	-5	-3	4	4	-2	-2	-2	-2	-2

(h) X_{2d}

Region	Ten initial chromosomes ($\times 5 \times 10^{-4}$)									
$Error \ge 3.0 \times 10^{-2}$	-5	-4	-6	-6	-6	-5	-5	-5	-5	-5
$10^{-2} \le Error < 3.0 \times 10^{-2}$	0	0	0	0	0	0	0	0	0	0
$Error < 10^{-2}$	2	2	2	2	2	2	2	2	2	2

Table 2. The parameters used in this implementation.

Name	Value
W_1	50
W_2	30
p_1	0.005
p_2	0.005
p_3	500 if $(0.4 - X_{01n}) > 0.0$ 200 if $(0.4 - X_{01n}) \le 0.0$
F_{limit}	-15 Nt